FINAL REPORT

ONR Project N00014-89-J-3109

MODELING AND COMPUTATIONAL ANALYSIS OF MULTISCALE PHENOMENA IN FLUID-STRUCTURE INTERACTION PROBLEMS

J. Tinsley Oden

Principal Investigator

The University of Texas

Texas Institute for Computational Mechanics

March, 1992

TABLE OF CONTENTS

Statement A per telecon
Dr. Philip Abraham ONR/Code 1132
Arlington, VA 22217-5000

NWW 4/14/92

Final Report

MODELING AND COMPUTATIONAL ANALYSIS OF MULTISCALE PHENOMENA IN FLUID-STRUCTURE INTERACTION PROBLEMS

1. Introductory Comments

This document summarizes research done during the period June 14, 1989 through December 14, 1991 on a project aimed at the development of advanced computational methods for modeling multiscale fluid-structure interaction phenomena.

The major thrust of the work reported is the development of the component parts of a new family of computational schemes that could be used to study large-scale problems in fluid-structure interaction. The overall theme of the project was the optimization of the computational process through the development of new adaptive techniques, data structures, a posteriori error estimates, and high-order solvers. The premise behind studying these types of high-order methods is that, with proper data structures, they could lead to exponential rates of convergence and thereby allow one to numerically solve very large problems, with features covering many different physical scales, using many fewer degreees of freedom than that required by conventional methods. In this spirit, the project was an ambitious one for it involved developing not only new adaptive algorithms for simulating the motion of elastic structures, but also techniques for fluid structure interaction and for modeling various regimes of incompressible viscous flows. Presumably, these methods could also be used to model the transition to turbulence and ultimately turbulent boundary layers on compliant surfaces.

Particular approaches explored during this project deviated significantly from many of the linearized or quasi-linearized approaches prevalent in the signal-processing literature. The theme here was to explore methods which attempted a direct assault on the governing equations. While this class of problems is exceedingly difficult, the payoff could be substantial in that not only would very general modeling capabilities be developed, but also very large simulations could conceivably be handled by new methodologies which were designed to optimize the computational process.

1.1 State of the Art at Start. The current state-of-the-art in computational modeling of large-scale subsonic flow-structure interaction problems is fairly well-known. Today a number of large-scale computer codes are available that are typically based on finite difference approaches. INS 3D, for example, a NASA--supported code developed for

incompressible flow simulation uses finite difference algorithms and splitting techniques to handle three-dimensional incompressible flow problems for Reynolds numbers of around 1,000. With the addition of various turbulence models, Reynolds numbers up to 10^6 or 10^8 can be handled, but the philosophy in resolving small-scale flow features is traditional: detailed features of the flow are modeled by incorporation of low order schemes involving many grid points. In complex flow calculations, the quality of a simulation is frequently measured in terms of the number of grid points in the mesh and efficiency is guaged by the number of iterations per grid point per second. Few of these calculations purport to model detailed features of boundary layers or transition to turbulence. Alternatively, spectral methods have been used in recent years with some success in modeling various turbulence phenomena. Unfortunately, these methods are primarily restricted to simple geometries and periodic boundary conditions and have not been used for significant interaction calculations.

The modeling of the structure itself and the fluid-structure interface also represents a class of problems that are at the cutting edge of computational mechanics. Contemporary methods are incapable of capturing sufficient detail to adequately model features of structure response that influence such phenomena as acoustics, or design of sensor locations on moving submerged vehicles, such as submarines, within a turbulent boundary layer. These types of modeling probems are still outside the reach of most conventional methods.

It has been known for a number of years that in certain simple cases, such as one-dimensional elliptic problems, a proper distribution of mesh parameters can produce exponential convergence of numerical simulations; that is to say, with a decrease of the mesh size or increase of the spectral order or proper location of grid points, the total approximation error may decay exponentially as the number of degrees of freedom are increased. This fact led to the question of whether or not a new family of algorithms could be developed which could optimize the distribution of various mesh parameters and lead to superaccurate methods which were capable of delivering high accuracy with very few degrees of freedom while capturing flow features that spanned many different spatial scales of resolution.

1.2 Goals of the Project at the Start. The research team had some experience developing hp-type finite element methods for simple heat conduction and viscous flow problems at the start of this project. With this background and with the possibility of high dividends if the problem of developing high-level hp-data structures could be adequately handled, the project was initiated and had as its initial goals the following:

1. Data structures. The development of new finite element data structures that allow for arbitrary mesh sizes by h-refinement and arbitrary spectral orders by p-enrichment. The data structure would have to be sufficiently general to be used in modeling not only the fluid but also the structure and the interface, and would be able to resolve multiple-scale features of flow phenomena through the use of hierarchical spectral-type approximations.

2. High-order temporal schemes. Since the time-dependent case was to be also modeled, it was necessary to develop very high-order temporal approximation schemes so that the order of the approximation in time would balance the corresponding higher-order approximation in space.

3. Elastic structure modeling. The same data structure used for modeling the fluid and the fluid interface was to be configured so as to be applicable to large-scale structural analysis as well.

4. New high-order methods for incompressible viscous flow. At the start of this project, no adaptive h-p methods had ever been developed for large-scale incompressible flow problems. A major goal was to also develop adaptive strategies that would automatically produce accurate results at a specified scales.

5. Fluid-structure interface. Various techniques for modeling the motion of a solid interface in viscous fluid had to be re-explored and adapted to these new approaches. This involved a new look at such classical methods as arbitrary Lagrange-Euler (ALE) methods, etc.

6. Iterative and flow solvers. It was realized at the onset for large-scale problems many of the direct methods for solving large linear systems would not be applicable to the subject class of problems. A parallel study on iterative solvers for large-scale h-p systems was also initiated. Among these solvers were those applicable to unsymmetrical systems since these occur in Navier-Stokes simulations.

7. A posteriori error estimation. The optimal control process and hp-adaptive schemes are based on the concept of manipulating mesh parameters so as to control the numerical error. Consequently, very reliable techniques for a posteriori error estimation are needed to drive these techniques. At the start of this project, a posteriori error estimates were available only for relatively simple linearly elliptic problems and, except for some very special cases, none had been developed and tested for Navier-Stokes equations or for general fluid structure interaction problems. The development of reliable error estimators schemes was identified as an important goal in this research effort.

2. Technical Approach

In the early stages of this project, attention was focused on three of the tasks listed in the previous section:

- Development of h-p data structures and adaptive strategies
- Development of high-order solvers for the incompressible Navier-Stokes equations
- Implementation of error estimation techniques that could be applicable

Our early work on data structures met with considerable success. A number of papers were published on the subject, and the techniques have been the subject of presentations at a number of national and international meetings, and extensions of the idea are even being used in one or two commercial software packages at the present. This data structure was, as far as is known by the author, the only data structure at the time that allowed a genuine h-p adaptive strategy in which the mesh size h and the local order p are actually treated as free parameters. The first such data structure was developed by the author and his Ph.D. students and is summarized in Appendix A, which is excerpted from a three-part paper on h-p data methods, written in 1989-90. Some of this work formed the basis of the doctoral dissertation of W. Rachowicz and was developed jointly by the author, L. Demkowicz, and W. Rachowicz.

The development high-order adaptive flow solvers for the Navier-Stokes equations presented a long series of difficulties. Not only is it necessary to develop robust and accurate, high-order time marching schemes, but also notorious difficulties with approximating pressures in incompressible flows had to be dealt with. Here reference is made to the so-called LBB condition in which approximations of the velocities cannot necessarily be made independently of the pressures. A poor pressure approximation can lead to unstable oscillations and, for higher order h-p schemes, these oscillations could be devastating.

For these reasons, a number of new high-order methods were first investigated. The first among these were the methods of characteristics. These schemes, based on rather classical ideas of characteristics, are used successfully in the European aerospace industry to solve subsonic flow problems. They are inherently low-order schemes, since they involve effectively a splitting of the convection and diffusion steps. However, these methods were easy to implement and were among the first to be coded and tried for two-dimensional cases employing our newly developed h-p data structure.

Initial results were deceptively good. Quite good results for transient two-dimensional incompressible flow problems were initially obtained. A lengthy series of

studies of techniques for pressure approximation was also initiated. It was finally decided that a penalty type approximation for pressures would be most appropriate for the h-p data structure.This involved attempts at developing a strategy that would always guarantee stable pressure approximation. As a general rule, this cannot be done with exterior penalty formulations unless reduced integration is used. The subject of reduced integration was studied extensively by the author from 1979 to 1982, but not for h-p approximations. This represented new ground and proved to be an exceedingly difficult issue. Some theoretical results were developed including a stability theorem that showed this: *if a Gaussian integration scheme of the order r is required to integrate the pressure approximation exactly, then the use of the scheme of order r--2 will always result in a stable pressure (i.e., the satisfaction of the LBB condition).* Mathematical proof of this condition for the two-dimensional Stokes problem was developed and numerous numerical experiments were done. This work was never published, because some of the numerical experiments led to data in contradiction to some of the theoretical results. Moreoever, the penalty approximation is notoriously difficult to stabilize. It was finally decided, after a number of months, that alternative techniques for pressure approximation should be investigated.

In addition to the method of characteristics, several new high-order methods of characteristics were developed. These were coded, experiments were done on a number of two-dimensional cases, and these methods were finally discarded as being expensive and often unreliable.

In the literature of the '80s, considerable discussion was given to Galerkin least-square methods and least-square methods for Navier-Stokes equations. A study of these techniques was also undertaken. These were also ultimately discarded because they were expensive, basically nonlinear in character, and often exhibited very high artificial viscosity that tended to obliterate important physical features of the solution.

More recently, a detailed study of high-order pressure-based schemes was undertaken. These schemes originate from a 1968 paper of A. Chorin and are perhaps the most popular methods in use today for solving the incompressible Navier-Stokes equations. They are based on the idea of first introducing a momentum step in which the velocity is advanced with or without a pressure term in the momentum equation. Then a Poisson equation for the pressure, or a pressure like variable, is solved and used to correct the velocities. These schemes are rich in mathematical history, having been studied by not only Chorin but also R. Temam and others. They are also the basis of a number of commercial codes.

Despite their popularity, pressure-based schemes are essentially first order accurate in time. There have been a number of papers that have attempted to extend these to second

order schemes, but they have been successful only for special types of boundary conditions. Investigators of the present project spent considerable effort in trying to develop high order pressure based schemes. These involve the use of implicit Runge-Kutta methods for the momentum step and the use of h-p adaptive solvers for the pressure step. To date, it has proved to be impossible to get a reliable second-order scheme. On the other hand, we did develop a robust first-order scheme that functions quite well on adaptive h-p meshes. While low-order accurate in time, the scheme nevetheless formed the basis of our numerical experimentation up until the conclusion of the project.

On another front, modeling of the elastic structure and the interface was also investigated. A high-order adaptive h-p algorithm was developed for linear elasticity problems. This scheme was also used in several experiments to model plates and shells with varying degrees of success. Also, some effort was spent on exploring methods for modeling moving interfaces. The result of these studies was a technique that was basically an Arbitrary Lagrange-Eulerian scheme, but it did include the capability of handling h-adaptivity. Plans were to eventually upgrade this interface scheme to h-p methods as well, but this phase of the study was never completed because of focus of the project on other issues.

In addition to the work on data structures and flow solvers for h-p adaptivity, a great deal of the project focused on the development new a-posteriori error estimators. The ability to quickly estimate the local, element-wise approximation error during the evolution of a calculation is a key feature of any successful adaptive scheme. At the initiation of this project, the available technology for error estimation was rather primitive being primarily confined to ad hoc methods for h adaptivity based on estimates of gradients of density or pressure; more rigorous techniques were successfully applied only to very limited classes of one-dimensional elliptic problems. During early stages of the project, considerable progress was made on the development of the error residual method, a concept first introduced by the present team in the early '80s. This method was further developed and applied to error estimation in linear elasticity problems and finally to Navier-Stokes equations. A paper detailing some of the earlier work on a posteriori error estimation for linear elliptic problems was produced early in the project and eventually implemented and applied to the Navier-Stokes equations.

More recently, a new variant of the error residual method was derived which is based on so-called flux balancing. It was discovered during the course of the work that the quality of an error estimator depends very much on the ability to approximate error fluxes at the boundaries of finite elements. Prior to this work, no error estimation technique was known that would function well for arbitrary h-p meshes. The basic idea behind these

techniques is this: the error in a gridcell depends upon the local (element-wise) residual which is a calculable function representing the degree to which the exact solution fails to satisfy the governing equations. In addition to this, the error is influenced by the boundary conditions on an element, the so-called flux of error from one cell into another. The idea behind the error residual method is to compute element residual and set up a local boundary value problem which is solved quickly for a running account of the local error. If the original problem is solved on a mesh for which polynomial approximation of degree p are used in a gridcell, the error estimate was generally done by estimating the error with polynomials of degree p or higher. Generally, a higher p was required for the error estimator in regions of singularities, such as in separation points, changes in boundary conditions or at point-line-singularities. It was discovered that for odd-order approximations, a key factor in the quality of the estimate was the balance of the error fluxes and a new technique was developed to resolve this issue. This work, recently accepted for publication in *Numerische Mathematik*, provides a significant advance to the a state-of-the-art posteriori error estimation and has proved to be successful in certain two-dimensional calculations of unsymmetrical elliptic systems.

Finally, some additional comments should be made concerning iterative solvers and direct solvers. A portion of the effort was spent investigating various types of iterative solvers with the understanding that large-scale fluid structure interaction problems cannot be efficiently handled by standard direct solvers or, induced, by direct solvers in general. This part of the work did not lead to any new methodologies but rather an assimilation, comparison, evaluation, and implementation of a number of existing schemes. Finally, in much of the work we continued to use a direct frontal solver because of its robustness and because the purpose of the early numerical experimentation was not necessarily to solve specific large-scale problems, but to explore the efficiency and functuality of new methods. However, we did implement and employ various iterative schemes such as the GMRES solver, Jacobi-Block iteration and other techniques. This work should prove to be valuable later when these techniques are extended to large-scale problems and to multi-processor environments.

3. Progress

3.1 Successes and Failures. In summarizing the approaches outlined in the previous section, one can point to a number of successes and a number of failures due to "blind alleys" pursued during the project. The major successes center around three areas: the development of new h-p adaptive strategies and data structures, the development of new methods for solving the incompressible Navier-Stokes equations and linear elasticity

equations using h-p adaptive solvers, and the development of a new family of a posteriori error estimators.

As noted above, a number of the initial approaches led to " blind alleys." These included in particular the lengthy studies of penalty approaches to h-p adaptive schemes for Navier-Stokes, the development of high order methods of characteristics, which while moderately successful did not appear to provide a robust foundation for further work, the exploration of Galerkin and least-squares based methods for primitive variable approximations of incompressible flow, and generally the development of high-order temporal schemes for Navier-Stokes equations. Indeed, this latter area remains open and represents one critical area in which considerable additional work needs to be done. At present, our experimental codes function on two classes of algortihms: 1) pressure based schemes, which employ a singly-implicit Runge-Kutta method for the momentum step, and the full high-order two-stage Runge-Kutta methods for the entire Navier-Stokes equations. There is no doubt that these schemes can be significantly improved with additional study.

A summary paper, including much of our work on data structures and some of our experimentation with high-order schemes in attached in Appendix B of this report.

4. Changes: Final Approaches and Final Goals

The success with the h-p adaptive schemes throughout the project made it clear that this is indeed a viable approach for large-scale fluid-structure interaction calculations. The data structures originally developed in this project have undergone a number of evolutionary changes and enough information has been gathered to produce new versions of these schemes which should be highly effective. The error estimation techniques developed seem to be quite promising, but additional work is needed to define their limitations and to extend them to truly coupled problems of fluid structure interaction and to rigorous methods of the full Navier-Stokes equations. The final stages of the project focused primarily on the Navier-Stokes simulation since the methods developed seem to work quite well for modeling structural response.

5. Relationship to Other Work

It should be noted that a number of the key components of the present project overlapped with on-going work supported in a parallel ONR project on structural acoustics. Indeed, much of the experiments developed on error estimation and on h-p data structures proved to be invaluable in developing similar procedures for BEM techniques for structural acoustics problems. In those problems, the primary modeling technique, however, was based on boundary element methods and somewhat different data structures and error

estimation techniques applied. On the other hand, experiments developed in both the fluid structure interaction effort and in the structural acoustics effort proved to be beneficial in each of the projects.

6. Problems Perceived

In future work, the direction toward more useful results is quite clear. Additional work on developing robust high-order solvers that function on high-order h-p meshes needs to be done. Continued work on pressure based schemes is intriguing and should be continued.

During the first year of this project we developed what appears to be the first h-p adaptive strategy and has been used successfully in the literature. This refers to a technique applicable to arbitrary h-p meshes, which when coupled with estimates of the local error, allows for a decision to be made whether to refine or coarsen the mesh, or enrich or de-enrich the spectral order. This is also a key to the success of h-p adaptive methods. Much additional work needs to be done on this subject. In recent months, a number of new adaptive strategies have been developed and have been implemented. These show in some cases, exponential convergence of the error versus CPU time, as opposed to degrees of freedom, and are regarded as significant advances in this subject. These methods also lend themselves to parallelization, and the implementation of these methods on parallel processers is suggested as a fundamental and important component of any additional work that is done on this subject.

It is believed that the basic components of an effective adaptive strategy for quite large problems in fluid structure interaction is in hand. Future efforts should include significant work on parallelization and, the development of load-balancing techniques for adaptive strategies. Indeed, while h-p adaptive methods or serial computers could very well represent the reduction of computational times by one to two orders of magnitude, the implementation of these schemes on multi-processor platforms could very well provide an additional order of magnitude or more in increases in efficiency.

As to major problems that are perceived, a significant problem may have to do with the stability and conditioning of h-p methods. With the increase in spectral order, one also finds an increase in the condition number of the underlying matrices. Progress has been made to develop pre-conditioning methods to control the stability of these schemes, but this is still a young area of research and much additional work needs to be done. The parallelization and preconditioning of adaptive h-p methods would clearly represent a fundamentally important area of study that needs to be done prior to the extension of these technologies to significant practical problems in modeling fluid-structure interactions.

7. Documentation

The research outlined above led to the publication of a number of articles, reports, and papers. Also several oral presentations of the work were made at national and international conferences and symposia. A summary of these documents and lectures is given below.

Articles

Oden, J.T., Bass, J.M., Huang, C-Y., and Berry, C.W., "Recent Results on Smart Algorithms and Adaptive Methods for Two- and Three-Dimensional problems in Computational Fluid Mechanics," *Computers and Structures*, 35(4), 1990, 381-396.

Strouboulis, T. and Oden, J.T., "A Posteriori Error Estimation of Finite Element Approximations in Fluid Mechanics," *Computer Methods in Applied Mechanics and Engineering*, 78, 1990, 201-242.

Oden, J.T., L. Demkowicz, W. Rachowicz, and T.A. Westermann, "A Posteriori Error Analysis in Finite Elements: The Element Residual Method for Symmetrizable Problems with Applications to Compressible Euler and Navier-Stokes Equations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 82, pp. 183-203, 1990.

Oden, J.T., L. Demkowicz, T. Liszka, and W. Rachowicz, "h-p Adaptive Finite Element Methods for Compressible and Incompressible Flows," *Computer Systems in Engineering*, Vol. 1, Nos. 2-4, pp. 523-534, 1990.

Oden, J.T., "Smart Algorithms and Adaptive Methods for Compressible and Incompressible Flow: Optimization of the Computational Process," *Very Large Scale Computation in the 21st Century*, Edited by J. P. Mesirov, SIAM Publications, Philadelphia, Chapter 7, pp. 87-99, 1991.

Oden, J.T. and L. Demkowicz, "h-p Adaptive Finite Element Methods in Computational Fluid Dynamics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 89, pp. 11-40, 1991.

Oden, J.T., T. Liszka, and W. Wu, "An h-p Adaptive Finite Element Method for Incompressible Viscous Flows," *The Mathematics of Finite Element and Applications VII*, Edited by J. Whiteman, Academic Press, Ltd., London, pp. 13-54, 1991.

Oden, J.T., "Theory and Implementation of High-Order Adaptive h-p Methods for the Analysis of Incompressible Viscous Flows," *Computational Nonlinear Mechanics in Aerospace Engineering*, AIAA Progress in Aeronautics and Astronautics Series, S.N. Atluri, ed. (in press).

Ainsworth, M. and Oden, J.T., "A Unified Approach to A Posteriori Error Estimation Using Element Residual Methods," *Numerische Mathematik*, (in press).

Reports

Oden, J.T., "Adaptive and Smart Algorithms for Fluid-Structure Interaction," Proceedings of the 28th AIAA Aerospace Sciences Meeting, Reno, NV, January, 1990.

Ainsworth, M. and J.T. Oden, "A Unified Approach to A Posteriori Error Estimation Using Element Residual Methods," TICOM Report 91-03, Austin, 1991.

Presentations

Oden, J.T., Demkowicz, L., Rachowicz, W. and Westermann, T.A., "A Posteriori Error Analysis in Finite Elements: The Element Residual Method for Symmetrizable Problems with Applications to Compressible Euler and Navier-Stokes Equations," Workshop on Reliability in Computational Mechanics, October 26-28, 1989, Austin, Texas.

Oden, J.T., "Progress in Fluid-Structure Interaction," ONR Solid-Fluid Interaction Accelerated Research Initiative Program Review, January 31-February 2, 1990.

Oden, J.T. and Bass, J.M., "New Developments in Adaptive Methods for Computational Fluid Dynamics," presented at the 9th International Conference on Computing Methods in Applied Sciences and Engineering, January 29-February 2, 1990, Paris, France

Oden, J.T., "Adaptive Finite Element Methods in Computational Fluid Mechanics," MSC's 1990 World User's Conference, March 29-31, 1990, Los Angeles, CA.

Oden, J.T., Liszka, T. and Wu, W., "An h-p Adaptive Finite Element Method for Incompressible Viscous Flows," presented at the 7th Conference on Mathematics of Finite Elements and Applications (MAFELAP), April, 1990, Brunel University, England.

Oden, J.T. and Demkowicz, L., "h-p Adaptive Finite Element Methods in Computational Fluid Dynamics," presented at the Second World Congress on Computational Mechanics, August, 1990, Stuttgart, West Germany.

Oden, J.T., "A General hp-Adaptive Finite Element Method for Broad Classes of Problems in Engineering," Fourth International Conference on Computing in Civil and Building Engineering, Tokyo, Japan, July 29-31, 1991.

Oden, J.T., "Progress on Adaptive High-Order hp-Finite Element Methods in Computational Fluid Dynamics," First U.S. National Congress, Chicago, IL, July 22-24, 1991.

Oden, J.T., "Smart Algorithms and Adaptive Finite Element Methods in CFD: Their Status and Potential," Second SA CFD Symposium, Stellenbosch, South Africa, June 24-27, 1991.

Oden, J.T., "Toward Optimal Control in Computational Fluid Dynamics: Smart Algorithms and Adaptive Methods," 27th Annual Meeting of the Society of Engineering Science, Santa Fe, New Mexico, October 21-25, 1990.

Oden, J.T., "h-p Adaptive Finite Element Methods for Compressible and Incompressible Flows," Symposium on Computational Technology for Flight Vehicles, NASA Langley Research Center, Hampton, VA, November 5-7, 1990.

Oden, J.T, "New Developments in Adaptive Finite Element Methods in Computational Fluid Dynamics," Sixth Chautauqua Meeting, White Plains, NY, September 9-11, 1990.

Oden, J.T., "Smart Algorithms and Adaptive Methods for Compressible and Incompressible Flow: Optimization of the Computational Process," presented at Boston, MA, October 1-3, 1990. Conference entitled, "Large-Scale Computing in the 21st Century."

Oden, J.T., "Progress in Fluid Structure Interaction: The Texas Group," ONR Fluid Structure Interaction Program Review Meeting, Austin, TX, January 24-25, 1991.

Oden, J.T., "Progress in Fluid Structure Interaction: The Texas Group," ONR Fluid Structure Interaction Program Review Meeting, Boston, MA, July 13-14, 1991.

APPENDIX A

Constrained Approximation: A General Approach to h-p Adaptive Data Structures

In the body of this report, refrence is made to various hp-data structures developed and studied during the course of this work. This appendix is provided to furnish some detail on one such data structure for completeness and for reference purposes. The particular formulation outlined here follows a paper by Demkowicz, Oden, Rachowicz, and Hardy [CMAME, Vol. 77, pp. 79–112] and, while not the most sophisticated version in use, provides all of the key details and features needed to understand the basic strategy.

1 h- and p-Adaptivity. Regular and Irregular Meshes

Let Ω be an open bounded domain in $\mathbb{R}^n$, $n = 2, 3$ with a sufficiently regular boundary $\partial\Omega$. In what follows, we shall restrict ourselves to a class of problems that can be formulated in the following abstract form:

$$\left.\begin{array}{l} \text{Find } \boldsymbol{u} \in \boldsymbol{X} \text{ such that} \\ \\ B(\boldsymbol{u}, \boldsymbol{v}) = L(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in \boldsymbol{X} \end{array}\right\} \tag{1.1}$$

Here:

$$\boldsymbol{X} = X \times X \cdots \times X (m \text{ times}) \tag{1.2}$$

where X a subspace of $H^1(\Omega)$, the Sobolev space of first order, $B(\cdot, \cdot)$ is a bilinear form on $\boldsymbol{X} \times \boldsymbol{X}$ of the following form

$$B(\boldsymbol{u}, \boldsymbol{v}) = \sum_{i,j=1}^{m} B_{ij}(u_i, v_j) \tag{1.3}$$

where $B_{ij}(\cdot, \cdot)$ are bilinear forms of scalar–valued arguments and $L(\cdot)$ is a linear form on $\boldsymbol{X}$ of the form

$$L(\boldsymbol{v}) = \sum_{j=1}^{m} L_j(v_j) \tag{1.4}$$

Finite Element Approximation

We assume that the domain Ω can be represented as a union of finite elements K_e, $e = 1,\ldots,M$. More precisely

$$\overline{\Omega} = \bigcup_{e=1}^{M} \overline{K}_e \tag{1.5}$$

and

$$\text{int } K_e \cap \text{ int } K_f = \emptyset \quad \text{for } e \neq f \tag{1.6}$$

Each of the elements K has a corresponding finite dimensional space of shape functions, denoted $X_h(K)$, for instance the space of polynomials of order p. The global finite element space X_h consists of functions which, when restricted to element K, belong to the local space of shape functions $X_h(K)$. Thus the global approximation is constructed by patching together the local shape functions in the usual way.

We shall adopt the fundamental requirement that the global approximation must be *continuous*. As we will see, this requirement leads to the notion of constrained approximations. Formally, the continuity assumption guarantees that the finite element space X_h is a subspace of $H^1(\Omega)$ and, with some additional assumptions if necessary, also a subspace of X. The approximate problem is easily obtained from (2.1) by substituting for $\boldsymbol{u}$ and $\boldsymbol{v}$ their approximations $\boldsymbol{u}_h$ and $\boldsymbol{v}_h$:

$$\left.\begin{array}{ll} \text{Find } \boldsymbol{u}_h \in \boldsymbol{X}_h & \text{such that} \\ B_h(\boldsymbol{u}_h, \boldsymbol{v}_h) = L_h(\boldsymbol{v}_h) & \forall \boldsymbol{v}_h \in \boldsymbol{X}_h \end{array}\right\} \tag{1.7}$$

Here

$$\boldsymbol{X}_h = X_h \times \ldots \times X_h (m \text{ times}) \tag{1.8}$$

which indicates that the *same* approximation has been applied to every component of $\boldsymbol{u}$. $B_h(\cdot,\cdot)$ and $L_h(\cdot)$ denote approximations to the original bilinear and linear forms resulting from numerical integration.

Adaptivity

A flow chart of a typical Adaptive Finite Element Method (AFEM) is shown in Fig. 1. The method consists of first generating an initial mesh and solving for the corresponding FEM approximate solution. Next, the error is estimated in some way and based on this (usually crude) approximation, one *adapts* the mesh, i.e., adds new degrees of freedom. The approximate problem for the new mesh is solved *again* and the whole procedure continues until certain error tolerances are met. Obviously, such a procedure requires an estimate of

the error over each element and a strategy to reduce the error by proper changes in the mesh parameters, h and p.

In the FEM the new degrees of freedom can be added in two different ways: elements may be locally refined or their spaces of shape functions may be enriched by incorporating new shape functions. As noted earlier, in the case of polynomials, this is done by increasing locally the degree of polynomials used to construct the shape functions, the first case being an h-refinement, and the second case a p-refinement. A combination of both is an (adaptive) h-p FEM. We remark that the process of increasing the local polynomial degrees for a fixed mesh size is mathematically akin to increasing the spectral order of the approximation and that, therefore, we also refer to h-p methods as "adaptive spectral–element" or "h-spectral" methods.

Regular and Irregular Meshes

As the result of local h-refinements, *irregular meshes* are introduced. Recall (see [6]) that a *node is called regular* if it constitutes a vertex for each of the neighboring elements; otherwise it is *irregular*. If all nodes in a mesh are regular, then the mesh itself is said to be *regular*. In the context of two–dimensional meshes, the maximum number of irregular nodes on an element side is referred to as the *index of irregularity*. Meshes with an index of irregularity equal one are called *1-irregular meshes*. The notion can be easily generalized to the three–dimensional case. (See [7] and literature cited therein for additional references.)

In the present work, we accept only 1-irregular meshes. In the two–dimensional context, this translates into the requirement that a "large" neighbor of an element may have no more than two "small" neighbors on a side; in the three–dimensional case, the number of neighbors sharing a side may go up to four, while the number of neighbors sharing an edge can be no more than two. This is frequently called the *"two-to-one"* rule (cf. [7]). Examples of regular and irregular meshes are shown in Fig. 2. There are several practical and theoretical reasons to accept only 1-irregular meshes, especially in the context of h-p methods. For a detailed argument, we refer to [8].

Our restriction to 1-irregular meshes imposes a simple restriction on the way any h-refinement can proceed: before an element is refined, a check for "larger" neighbors must be made. If any such neighbors exist, they must be refined first and only then can the element in question be refined.

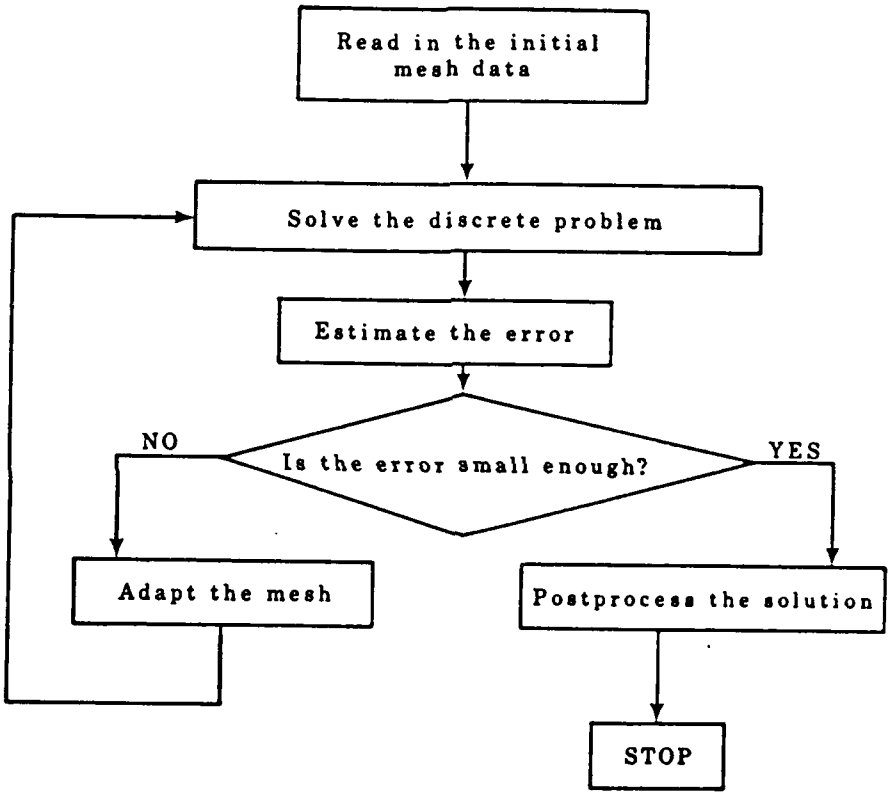


Figure 1: Typical flow chart of an adaptive method

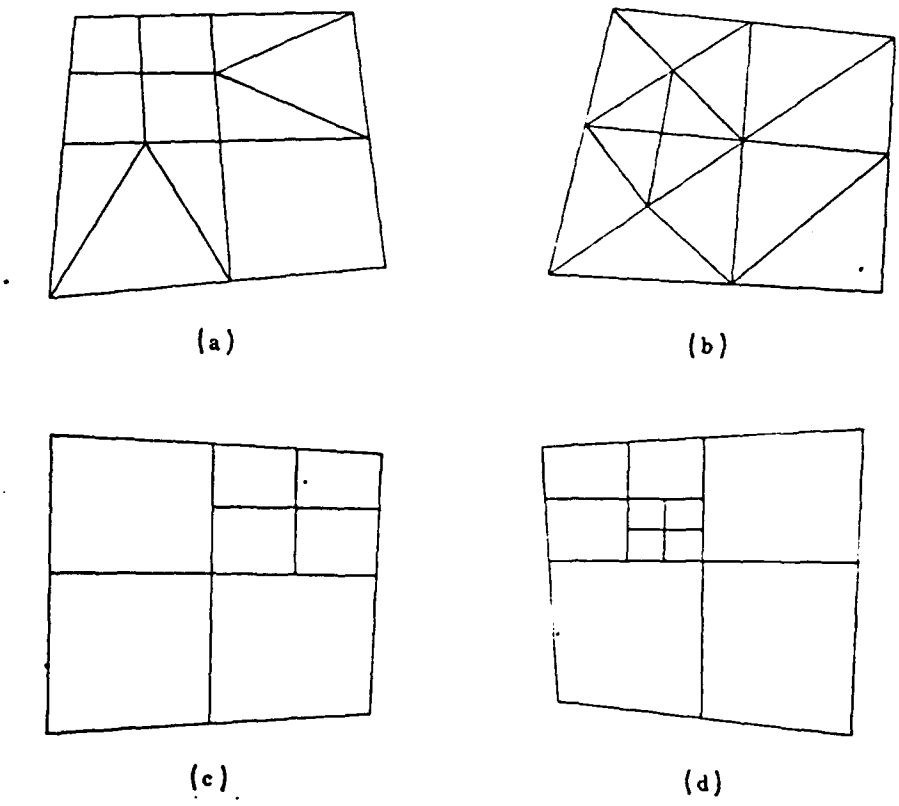


Figure 2: Examples of regular and irregular meshes: (a) and (b) — regular mesh; (c) 1-irregular mesh (index of irregularity = 1); (d) 2-irregular mesh.

Continuity and Constrained Approximation

The presence of irregular nodes makes the handling of the continuity assumption non-standard and leads to the notion of a constrained approximation. As an example, consider a mesh of three rectangular Q^2 elements (quadrilateral elements with biquadratic shape functions) with standard Lagrange degrees of freedom, as shown in Fig. 3. Clearly, a function u_h defined over these elements need not to be continuous across element interfaces due to the presence of two irregular nodes, A and B. For instance, the values of u_h on side CDE of element K are determined uniquely by (say) its values at points C, D and E, while those on side CD of element K_2 are defined by specifying degrees of freedom in K. In order to make u_h continuous, the value of u_h at A from the side of K_2 must be *forced* to be equal to the value of u_h at A from the side of K_1, which is equivalent to the elimination of the degree of freedom associated with point A by enforcing the constraint

$$u_h(A) = \alpha u_h(C) + \beta u_h(D) + \gamma u_h(E) \tag{1.9}$$

with proper coefficients α, β and γ. The situation is much more complicated in the case of *different* orders of approximation within each element, and we discuss it in detail in Section 3.

2 Constrained Approximation

In this section, we develop the general concept of constrained approximations. We shall see in particular the impact of the constraints on such basic ingredients to the FEM as element stiffness matrix and load vector calculations.

Approximation on the Element Level

Let K be a finite element with the corresponding space of shape functions

$$X_h(K).$$

The set Q_K of element degrees of freedom N_K, as usual, is viewed as a set of linear functionals defined on $X_h(K)$. We assume that the set Q_K of degrees of freedom

$$\{\varphi_{i,K} : X_h(K) \to \mathbb{R} \mid i \in N_K \subset N\} \tag{2.1}$$

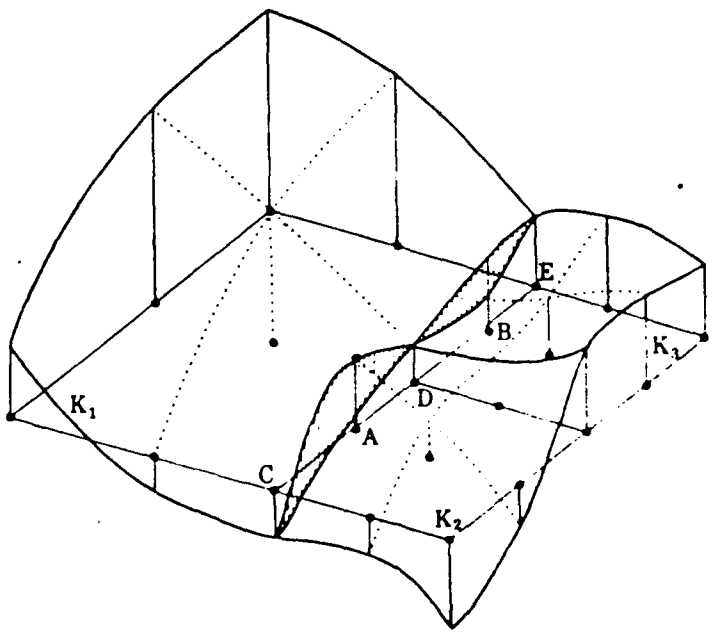


Figure 3: Example of an unconstrained, discontinuous approximation.

is $X_h(K)$-unisolvent (see [9, 10]), i.e.,

1° $\varphi_{i,K}, i \in N_K$, are linearly independent

2° $\bigoplus_i \mathbb{R}\varphi_{i,K}, i \in N_K$ is a dual space to $X_h(K)$

As indicated, we assign to each linear functional $\varphi_K \in Q_K$ an integer label i, $\varphi \to \varphi_{i,K}$ and denote by N_K the set of such labels for element K. The *element shape functions* $\chi_{j,K}$ are *defined* as a dual basis to $\varphi_{i,K}$, i.e.,

$$< \varphi_{i,K}, \chi_{j,K} >= \delta_{ij} \qquad i,j \in I\!N_K \tag{2.2}$$

and the finite element approximation u_h within the element K is of the form

$$u_h = \sum_{i \in N_k} u_h^i \chi_{i,K}$$

where $u_h^i = \varphi_{i,K}(u_h)$.

In what follows, we restrict ourselves to Lagrange– and Hermite–types of degrees of freedom. In other words, we assume that each of the functionals φ_i is of the form

$$\varphi \ : \ u \to D_{\boldsymbol{x}}^k u(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k) \ k = 0, 1, \ldots \tag{2.3}$$

where $D_{\boldsymbol{x}}^k u$ denotes the k-th order differential of u evaluated at point $\boldsymbol{x}$ (as usual, $D_{\boldsymbol{x}}^0 u = u(\boldsymbol{x})$). Vectors $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k$ at this point denote arbitrary vectors in $\mathbb{R}^n$, $n = 2, 3$. Thus, every degree of freedom can be identified with a point $\boldsymbol{x}$ and k vectors $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k$. (Later on, in the context of subparametric finite elements, we shall accept functionals which are linear combinations of functionals of the form (3.3).)

Construction of the Unconstrained Finite Element Space $\widetilde{X}_h$

We introduce the following formal definition of the *unconstrained finite element space* $\widetilde{X}_h$. A function u_h: $\Omega \to \mathbb{R}$ belongs to $\widetilde{X}_h$ if and only if the following two conditions are satisfied:

1. The restriction of u_h to an element K belongs to the local space of shape functions, i.e.,

$$u_h|_K \in X_h(K) \qquad \forall K \tag{2.4}$$

2. For every pair of elements K_e and K_f and corresponding pair of degrees of freedom

$$\varphi_{K_e} \ : \ X_h(K_e) \rightarrow I\!R$$

$$\varphi_{K_f} \ : \ X_h(K_f) \rightarrow I\!R$$

such that φ_{K_e} and φ_{K_f} are defined by (3.3) through the *same, common* point $\boldsymbol{x}$ and vectors $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K$, the two degrees of freedom take on the *same* value on u_h when restricted to K_e or K_f respectively, i.e.,

$$\varphi_{K_e}(u_h|_{K_e}) = \varphi_{K_f}(u_h|_{K_f}) \tag{2.5}$$

Note that the elements of $\widetilde{X}_h$ need not be continuous (see. Fig. 3).

Global Degrees of Freedom

Due to the const ıction of the space $\widetilde{X}_h$, we can introduce the global degrees of freedom identified by points (nodes) $\boldsymbol{x}$ and vectors $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k$. We define the linear functional Φ on $\widetilde{X}_h$,

$$\Phi : \widetilde{X}_h \rightarrow I\!R, \Phi(u_h) = \varphi_K(u_h|_K), \tag{2.6}$$

where K is an element with the corresponding degree of freedom identified with point $\boldsymbol{x}$ and vectors $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k$. Note that due to the definition of $\widetilde{X}_h$, the global degrees of freedom are well defined.

The Unconstrained Base Functions

The unconstrained base functions $\tilde{e}_i$ are introduced as a dual basis to the space of the global degrees of freedom, i.e.,

$$< \Phi_j, \tilde{e}_i >= \delta_{ij} \tag{2.7}$$

Note that $\tilde{e}_i$ may be discontinuous.

Construction of the Constrained Finite Element Space X_h

At this point, somewhat arbitrarily, we divide all global degrees of freedom into two subsets: active and constrained. We use the notation,

- the set N^a of indices assigned to active degrees of freedom and
- the set N^c of indices assigned to constrained degrees of freedom

By an "active" degree of freedom, we mean one of a set of linearly independent functionals that define the parameters associated with a global stiffness matrix for the problem at hand; "constrained" degrees of freedom are linear combinations of active degeres of freedom defined by constraints associated with element connectivity.

We assume that for each constrained degree of freedom Φ_i, $i \in N^c$, there exists a set $I(i)$ of corresponding active degrees of freedom, $I(i) \subset N^a$, and a vector R_{ij}, $j \in I(i)$, such that the following equality holds:

$$\Phi_i(u_h) = \sum_{j \in I(i)} R_{ij} \Phi_j(u_h) \tag{2.8}$$

We now introduce the constrained finite element space X_h as

$$X_h = \left\{ u_h \in \widetilde{X}_h \mid \Phi_i(u_h) = \sum_{j \in I(i)} R_{ij} \Phi_j(u_h) \qquad \forall i \in N^c \right\} \tag{2.9}$$

Assuming that the constraints are linearly independent, we see that X_h is dual to the space spanned by only active degrees of freedom. As usual, we define the base functions e_j, $j \in N^a$, as a dual basis to the set of active degrees of freedom:

$$e_j \in X_h, \; < \Phi_i, e_j >= \delta_{ij} \quad i, j \in N^a \tag{2.10}$$

Though, at this point, the choice of constrained degrees of freedom is arbitrary, we implicitly assume that with the proper choice of constraints the resulting finite element space X_h consists of only continuous functions.

Relation Between Unconstrained and Constrained Base Functions

Let u_h be an arbitrary function belonging to X_h. Then u_h must be of the following form:

$$\begin{aligned} u_h &= \sum_{i \in N^a} u_i \tilde{e}_i + \sum_{j \in N^c} u_j \tilde{e}_j \\ &= \sum_{i \in N^a} u_i \tilde{e}_i + \sum_{j \in N^c} \sum_{k \in I(j)} R_{jk} u_k \tilde{e}_j \end{aligned} \tag{2.11}$$

Introducing for every $i \in N^a$ the set

$$S(i) = \{j \in N^c \mid i \in I(j)\} \tag{2.12}$$

we rewrite u_h in the form

$$\begin{aligned} u_h &= \sum_{i \in N^a} u_i \tilde{e}_i + \sum_{k \in N^a} \sum_{j \in S(k)} u_h R_{jk} \tilde{e}_j \\ &= \sum_{i \in N^a} u_i \left(\tilde{e}_i + \sum_{j \in S(i)} R_{ji} \tilde{e}_j \right) \end{aligned} \tag{2.13}$$

We claim that functions

$$e_i = \tilde{e}_i + \sum_{j \in S(i)} R_{ji} \tilde{e}_j \qquad i \in N^a \tag{2.14}$$

form the dual basis to functionals Φ_i, $i \in N^a$. Indeed

$$< \Phi_j, e_i > = < \Phi_j, \tilde{e}_i > + \sum_{k \in S(i)} R_{ki} < \Phi_j, \tilde{e}_k > = \delta_{ij} \tag{2.15}$$

since $S(i) \subset N^c$.

Calculation of the Global Load Vector and Stiffness Matrix

For simplicity, we restrict ourselves to the case of a single equation. In the case of systems, the same procedure is applied for every linear form (2.7) and bilinear form (2.4).

Substituting (3.14) into both sides of (2.10), we obtain formulas for the load vector and stiffness matrix:

$$L_h(e_i) = L_h(\tilde{e}_i) + \sum_{k \in S(i)} R_{ki} L_h(\tilde{e}_k) \tag{2.16}$$

$$\begin{aligned} B_h(e_i, e_j) &= B_h(\tilde{e}_i, \tilde{e}_j) \\ &+ \sum_{k \in S(i)} R_{ki} B_h(\tilde{e}_k, \tilde{e}_j) \\ &+ \sum_{l \in S(j)} R_{li} B_h(\tilde{e}_i, \tilde{e}_l) \\ &+ \sum_{k \in S(i)} \sum_{l \in S(j)} R_{ki} R_{lj} B_h(\tilde{e}_k, \tilde{e}_l) \end{aligned} \tag{2.17}$$

Element Level Revisited—Modified Element Stiffness Matrix and Load Vector

Consider an element K. Let $N^a(K)$ and $N^c(K)$ denote sets of indices corresponding to active and constrained degrees of freedom for the element. Assuming that, as usual, the load vector and stiffness matrix are calculated by summing up the contributions of all elements, i.e.,

$$L_h(u_h) = \sum_K L_{h,K}(u_h|_K) \tag{2.18}$$

and

$$B_h(u_h, v_h) = \sum_K B_{h,K}(u_h|_K, v_h|_K) \tag{2.19}$$

we arrive at the practical issue of how to calculate the contributions of element K to the global load vector and stiffness matrix. Toward resolving this question, we introduce

- the usual element load vector

$$\tilde{b}_{i,K} = L_{h,K}(\chi_{i,K}) \quad i \in N^a(K) \cup N^c(K) \tag{2.20}$$

- the element stiffness matrix

$$\tilde{B}_{ij,K} = B_{h,K}(\chi_{i,K}, \chi_{j,K}) \quad i,j \in N^a(K) \cup N^c(K) \tag{2.21}$$

- the set of associated *active* degrees of freedom for element K

$$N(K) = N^a(K) \cup \bigcup_{j \in N^c(K)} I(j) \tag{2.22}$$

Notice that the two sets on the right–hand side of (3.22) need not be disjoint!

- the element contribution to the global load vector (modified element load vector)

$$b_{i,K} = L_{h,K}(e_i|_K) \quad i \in N(K) \tag{2.23}$$

- the element contribution to the global stiffness matrix (modified element stiffness matrix)

$$B_{ij,K} = B_{h,K}(e_i|_K, e_j|_K) \quad i,j \in N(K) \tag{2.24}$$

Algorithms for the Calculation of $b_{i,K}$ and $B_{ij,K}$

The load vector and stiffness matrix for the constrained degrees of freedom are now computed by transformations embodied in the following sample algorithms:

FOR $i \in N(K)$

$b_{i,K} = 0$

ENDFOR

FOR $i \in N^a(K) \cup N^c(K)$

IF $i \in N^a(K)$ THEN

$b_{i,K} = b_{i,K} + \tilde{b}_{i,K}$

ENDIF

IF $i \in N^c(K)$ THEN

FOR $k \in I(i)$

$[b_{k,K} = b_{k,K} + R_{ik}\tilde{b}_{i,K}]$

ENDFOR

ENDIF

ENDFOR

The $B_{ij,K}$ Algorithm

```
FOR                 i,j ∈ N(K)
        B_{ij,K} = 0
ENDFOR

FOR                 i,j ∈ N^a(K) ∪ N^c(K)
        IF          i ∈ N^a(K) AND j ∈ N^a(K) THEN
                    B_{ij,K} = B_{ij,K} + B̃_{ij,K}
        ENDIF

        IF          i ∈ N^c(K) AND j ∈ N^a(K) THEN
                    FOR k ∈ I(i)
                    B_{kj,K} = B_{kj,K} + R_{ik} B̃_{kj,K}
                    ENDFOR
        ENDIF
        IF          i ∈ N^a(K) AND j ∈ N^c(K) THEN
                    FOR l ∈ I(j)
                    B_{il,K} = B_{il,K} + R_{jl} B̃_{il,K}
                    ENDFOR
        ENDIF

        IF          i ∈ N^c(K) AND j ∈ N^c(K) THEN
                    FOR k ∈ I(i), l ∈ I(j)
                    B_{kl,K} = B_{kl,K} + R_{ik} R_{jl} B̃_{kl,K}
                    ENDFOR
        ENDIF
ENDFOR
```

3 An h-p Adaptive Finite Element Method

This section is devoted to a presentation of an h-p Adaptive Finite Element Method which provides for instantaneous h- and p-refinements and unrefinements. For simplicity, the discussion is restricted to the two–dimensional case.

We shall adopt the following assumptions:

- the initial mesh is (topologically) a portion of a regular, rectangular grid in $\mathbb{R}^2$;
- only 1-irregular meshes are accepted for all h-refinements;

- the local order of approximations may differ in each element;
- the approximation must be continuous.

One-Dimensional Hierarchical Element of an Arbitrary Order p

We begin our discussion with the definition of the standard one–dimensional hierarchical master element defined on the interval $[-1, 1]$.

The element degrees of freedom are identified with three nodes: two endpoints and the midpoint $x = 0$. If the element is of the first order, only two degrees of freedom are present — these are the nodal values at $x = -1$ and $x = 1$. Starting with $p = 2$, new degrees of freedom are added which are of the form

$$\varphi_p(u) = \frac{1}{\lambda_p} \frac{d^p u}{dx^p}(0) \tag{3.25}$$

where λ_p is a scaling factor.

The corresponding shape functions are illustrated in Fig. 4.

Definition of the Hierarchical Square Q^p Master Element

Setting $\widehat{K} = [-1, 1] \times [-1, 1]$, we define the space of shape functions as

$$X_h(\widehat{K}) = Q^p(\widehat{K}) \tag{3.26}$$

where Q^p denotes the space of polynomials up to p-th order with respect to each of the variables separately.

The degrees of freedom are defined as follows:

- function values at four vertices:

$$u(-1, -1), u(1, -1), u(1, 1), u(-1, 1) \tag{3.27}$$

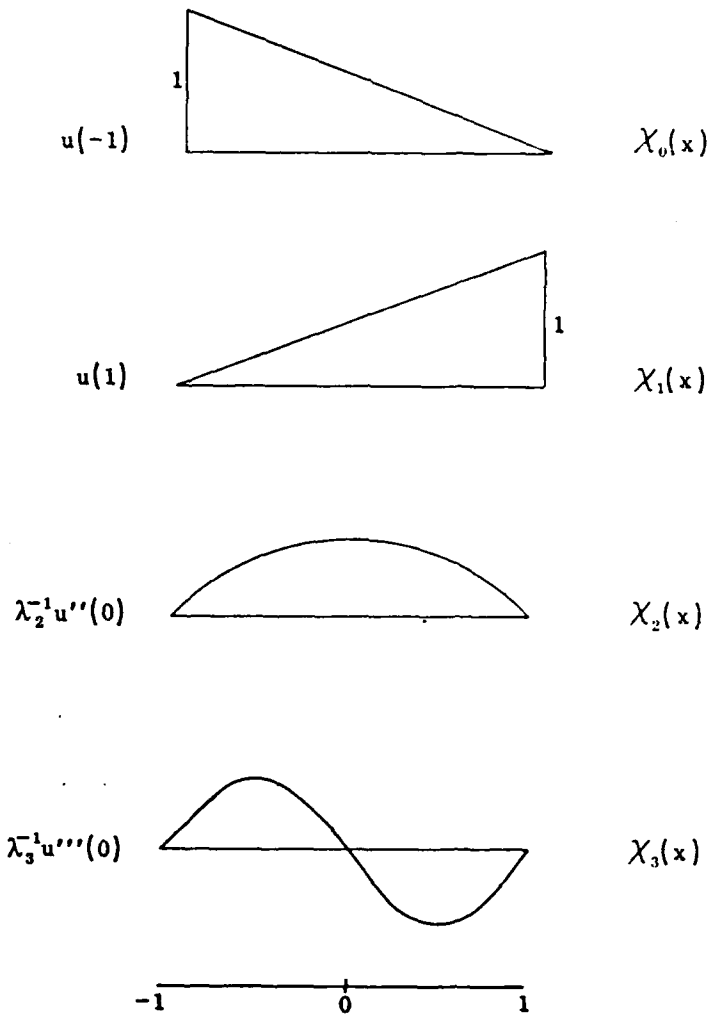


Figure 4: One–dimensional hierarchical master element. Degrees of freedom and corresponding shape functions.

- tangential derivatives (up to a multiplicative constant) up to p-th order associated with the midpoints of the four edges:

$$\begin{aligned}
&\lambda_k^{-1}\frac{\partial^k u}{\partial x^k}(0,-1) \quad k=2,\ldots,p \\
&\lambda_k^{-1}\frac{\partial^k u}{\partial y^k}(1,0) \quad k=2,\ldots,p \\
&\lambda_k^{-1}\frac{\partial^k u}{\partial x^k}(0,1) \quad k=2,\ldots,p \\
&\lambda_k^{-1}\frac{\partial^k u}{\partial y^k}(-1,0) \quad k=2,\ldots,p
\end{aligned} \tag{3.28}$$

- mixed order derivatives associated with the central node

$$\lambda_k^{-1}\lambda_\ell^{-1}\frac{\partial^{k+l} u}{\partial x^k \partial y^\ell}(0,0) \quad k,l=2,\ldots,p \tag{3.29}$$

One can easily see that the space of the shape functions $Q^p(\widehat{K})$ is a tensor product of $P^p[-1,1]$ with itself and the degrees of freedom just introduced are simply the tensor products of the degrees of freedom for the one–dimensional element. More precisely, if $u \in Q^p$, then u is of the form

$$u(x,y) = \sum_k u^k v_k(x) w_k(y) \tag{3.30}$$

where $v_k, w_k \in P^p(-1,1)$, and each of the degrees of freedom can be represented in the form

$$(\varphi_i \otimes \varphi_j)(u) \;=\; \sum_k u^k (\varphi_i \otimes \varphi_j)(v_k \otimes w_k) \tag{3.31}$$

This in particular implies that the corresponding shape functions can be identified with the tensor products of one–dimensional shape functions, which are of the form

$$\chi_i(x)\chi_j(y) \quad i,j=0,1,\ldots,p \tag{3.32}$$

For $i,j=0,1$, we have the usual bilinear element with four nodal degrees of freedom.

Master Element of an Enriched Order

Let $\widehat{K}$ be an element of p-th order. By adding additional shape functions corresponding to the (p + 1)th order, together with the corresponding degrees of freedom, we obtain a well-defined finite element whose space of shape functions includes Q^p. In particular, by adding all the additional degrees of freedom corresponding to the (p + 1)th order, we pass to the element Q^{p+1} *without modifying* the existing shape functions and degrees of freedom of Q^p. Equivalently, we can begin an adaptive process with element Q^{p+1} and eliminate some degrees of freedom to pass to an incomplete element of a lower order.

Subparametric Hierarchical Elements

Consider the master element of (possibly) an incomplete order p. Even though p can be arbitrarily large, the element may be only Q^1-complete, which means that some of the nodes may be missing. An example of such an element is presented in Fig. 5. An arbitrary location of the seven nodes in the plane (x, y) determine uniquely a map T from the master element into $\mathbb{R}^2$, the components of T belonging to the incomplete Q^2 space. More precisely, if $\psi_i, i = 1,\ldots,9$ are the regular shape functions for the nine node biquadratic element, then

$$T(\hat{x}, \hat{y}) = \sum_{i=1}^{9} a_i \psi_i(\hat{x}, \hat{y}) \tag{3.33}$$

with the assumption that $a_6 = \frac{1}{2}(a_2 + a_3)$ and $a_7 = \frac{1}{2}(a_3 + a_4)$.

We have the classical definition of the subparametric element,

$$K = T(\widehat{K}) \tag{3.34}$$

with the space of test functions defined as

$$X_h(K) = \left\{ u = \hat{u} \cdot T^{-1} \mid \hat{u} \in X_h(\widehat{K}) \right\} \tag{3.35}$$

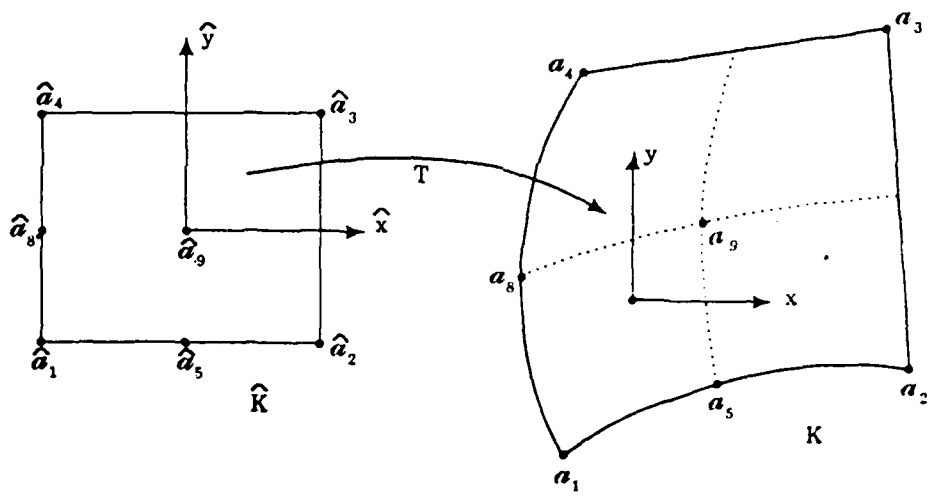


Figure 5: Concept of the subparametric element.

and the degrees of freedom

$$< \varphi, u >=< \hat{\varphi}, \hat{u} > \text{ where } u = \hat{u} \circ T^{-1} \tag{3.36}$$

Interpretation of the Degrees of Freedom

The degrees of freedom associated with vertices are simply the function values evaluated at these points. The degrees of freedom associated with the midpoints of element edges and central nodes are more complicated. It follows from definition (4.36) that they may be interpreted as certain *linear combinations* of directional derivatives. The form of directional or mixed derivatives appropriate for the master element is preserved only if the map T is linear. Thus, it is understood that the degrees of freedom discussed in the previous section should be interpreted broadly enough to include linear combinations of these Hermite-type degrees of freedom.

Continuity for Regular Meshes

One of the fundamental advantages of using the hierarchical shape functions is the ease with which they allow one to construct a continuous approximation with locally variable order of approximation. A typical situation is illustrated in Fig. 6. If elements K_1 and K_2 are to support polynomials of degree, say, one and three, respectively, then there are at least two ways to enforce continuity across the interelement boundary. One way is to add two extra shape functions of second and third order corresponding to the middle node A of element K_1. Alternatively, the same two shape functions may be deleted from element K_2. One can, of course, establish a "golden rule" method by adding one extra shape function of second order to K_1, and deleting the third order shape function from K_2. In all these cases, a *common order of approximation* along the interelement boundary can be enforced by simply adding or deleting the respective shape functions from the neighboring elements. While any of these choices can be made, the results described here employ the "maximum rule" in which the higher–order approximation dominates lower orders. Thus, if an element is p-refined, i.e., a higher order approximation of degree $\overline{p}$ is introduced, the neighbors of lower order are enriched by the addition of extra shape functions of degree $\overline{p}$ necessary to guarantee continuity of the approximation.

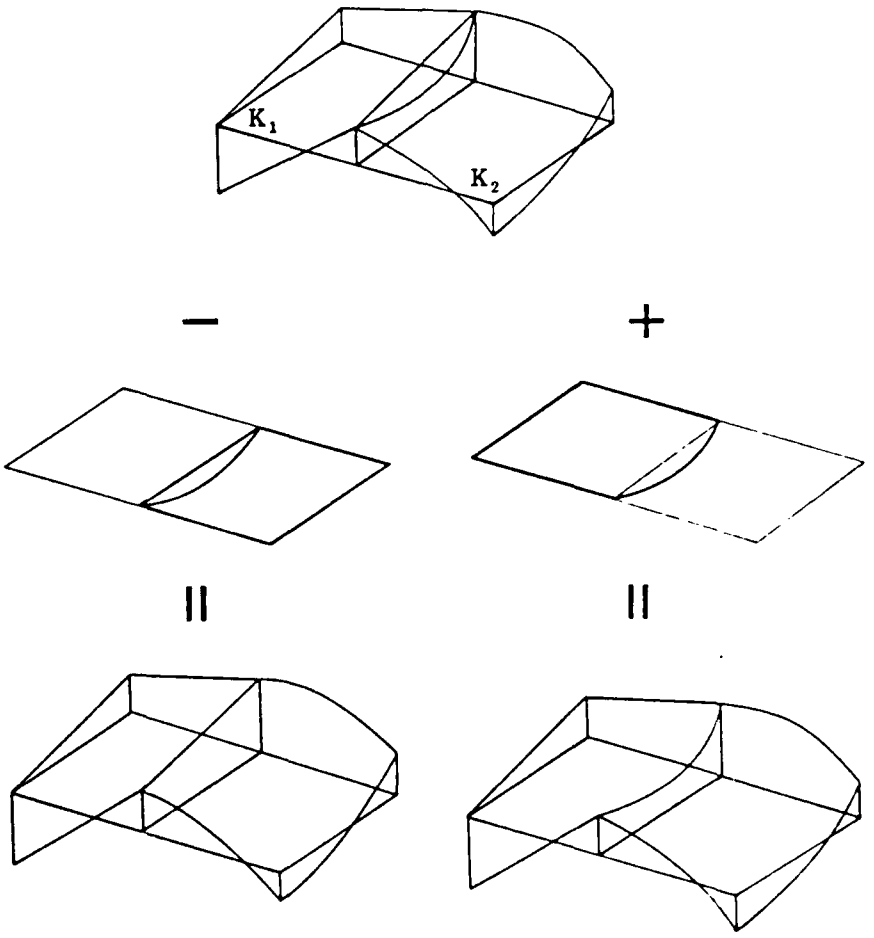


Figure 6: Continuity by hierarchical shape functions.

Constraints in the One-Dimensional Case

In the case of irregular meshes, continuity has to be enforced by means of the constrained approximation. To fix ideas, consider the generic, one-dimensional case shown in Fig. 7. The approximation on the small elements $[-1,0]$ and $[0,1]$ must match the approximation on the large element $[-1,1]$.

We first choose the scaling factors λ_p in (4.25) in such a way that the corresponding shape functions for the one-dimensional master element have the following form

$$\left.\begin{aligned} \chi_0 &= \frac{1}{2}(1-\xi) \\ \chi_1 &= \frac{1}{2}(1+\xi) \\ \chi_p(\xi) &= \begin{cases} \xi^p - 1 & p = 2,4,6,\ldots \\ \xi^p - \xi & p = 3,5,7,\ldots \end{cases} \end{aligned}\right\} \tag{3.37}$$

Assume next that all degrees of freedom for the large element are active. The question is: what degrees of freedom must the small elements take on in order that the functions supported on the two small elements exactly coincide with shape functions of the large element?

From the fact that (4.25) is a dual basis to (4.37), we get

$$\varphi_p(\chi_p) = \frac{1}{\lambda_p} p! = 1 \quad p = 2,3,\ldots \tag{3.38}$$

and therefore $\lambda_p = p!$.

The transformation map from $[-1,1]$ onto $[-1,0]$ is of the form

$$x = -\frac{1}{2} + \frac{1}{2}\xi \tag{3.39}$$

with inverse $\xi = 2x + 1$. This yields the following formulas for the shape functions ${}^{\ell}\chi_p, p =$

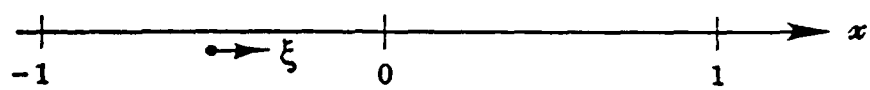


Figure 7: Derivation of the constraint coefficients.

$0, 1, 2, \ldots$ For the (left-hand side) element $[-1, 0]$ (recall definition (4.35)).

$$\begin{aligned} {}^{\ell}\chi_0(x) &= -x \\ {}^{\ell}\chi_1(x) &= x + 1 \\ {}^{\ell}\chi_p(x) &= 1 - (2x+1)^p \quad p = 2, 4, 6, \ldots \\ {}^{\ell}\chi_p(x) &= (2x+1)^p - (2x+1) \quad p = 3, 5, 7, \ldots \end{aligned} \tag{3.40}$$

and the corresponding formulas for the degrees of freedom are (recall (4.36)).

$$\begin{aligned} <^{l} \varphi_0, u > &= u(-1) \\ <^{l} \varphi_1, u > &= u(0) \\ <^{l} \varphi_p, u > &= \frac{1}{2^p p!} \frac{d^p u}{dx^p} \left(-\frac{1}{2} \right) \quad p = 2, 3, \ldots \end{aligned} \tag{3.41}$$

Now let $u(x)$ ($x = \xi$ for the master element) be any function defined by the shape functions on $[-1, 1]$, i.e.,

$$u(x) = \sum_{q=0}^{k} \varphi_q(u) \chi_q(x) \tag{3.42}$$

In order to represent $u(x)$ for $x \in [-1, 0]$ in terms of the shape functions on $[-1, 0]$, we have to calculate the values of the degrees of freedom (4.41). We get

$$\begin{aligned} <^{\ell} \varphi_0, u > &= \varphi_0(u) <^{\ell} \varphi_0, \chi_0 > + \sum_{q=1}^{k} \varphi_q(u) <^{l} \varphi_0, \chi_q > \\ &= < \varphi_0, u > \\ <^{l} \varphi_1, u > &= \varphi_0(u) <^{l} \varphi_1, \chi_0 > + \varphi_1(u) <^{l} \varphi_1, \chi_1 > \\ &+ \sum_{q=2}^{k} \varphi_q(u) <^{l} \varphi_1, \chi_q > \\ &= \frac{1}{2} < \varphi_1, u > + \sum_{q=2}^{k} {}^{l}R_{q1} < \varphi_q, u > \end{aligned} \tag{3.43}$$

where

$$\begin{aligned} {}^{l}R_{q1} &= <^{l} \varphi_1, \chi_q > \\ &= \begin{cases} 1 & \text{if } q \text{ is even} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3.44}$$

For $p \geq 2$,

$$
\begin{aligned}
<^l \varphi_p, u> &= \varphi_0(u) <^l \varphi_p, \chi_0> + \sum_{q=1}^{k} \varphi_q(u) <^l \varphi_p, \chi_q> \\
&= 0 + \sum_{q=1}^{k} {}^lR_{qp} < \varphi_q, u>
\end{aligned}
$$

where

$$
\begin{aligned}
{}^lR_{qp} &= <^l \varphi_p, \chi_q> \\
&= \begin{cases} 0 & \text{for } q < p \\ \dfrac{1}{2^q}(-1)^{p+q}\binom{q}{p} = \dfrac{(-1)^{p+q}}{2^q}\dfrac{q!}{p!(q-p)!} & \text{for } q \geq p \end{cases}
\end{aligned}
\tag{3.45}
$$

The same procedure applied to the right-hand side element [0,1] yields the following:

The transformation from $[-1,1]$ *onto* $[0,1]$

$$x = \frac{1}{2} + \frac{1}{2}\xi \tag{3.46}$$

with inverse $\xi = 2x - 1$.

The shape functions ${}^r\chi_p, p = 0,1,2,\ldots$:

$$
{}^r\chi_0(x) = 1 - x \tag{3.47}
$$

$$
{}^r\chi_1(x) = x \tag{3.48}
$$

$$
{}^r\chi_p(x) = \begin{cases} 1 - (2x-1)^p & p \text{ even} \\ (2x-1)^p - (2x-1) & p \text{ odd} \end{cases} \tag{3.49}
$$

The degrees of freedom ${}^r\varphi_p$:

$$
\begin{aligned}
< {}^r\varphi_0, u > &= u(0) \\
< {}^r\varphi_1, u > &= u(1) \\
< {}^r\varphi_p, u > &= \frac{1}{2^p p!}\frac{d^p u}{dx^p}\left(\frac{1}{2}\right)
\end{aligned}
\tag{3.50}
$$

The constraints

$$< {}^{r}\varphi_0, u >= \frac{1}{2} < \varphi_0, u > + \frac{1}{2} < \varphi_1, u > + \sum_{q=2}^{k} {}^{r}R_{q0} < \varphi_q, u > \tag{3.51}$$

where

$${}^{r}R_{q0} = \begin{cases} 1 & \text{if } q \text{ is even} \\ 0 & \text{otherwise} \end{cases} \tag{3.52}$$

$$< {}^{r}\varphi_1, u >=< \varphi_1, u > \tag{3.53}$$

and for $p \geq 2$

$$< {}^{r}\varphi_p, u >=< \sum_{q=2}^{k} {}^{r}R_{qp} < \varphi_q, u > \tag{3.54}$$

where

$${}^{r}R_{qp} = {}^{r}\varphi_p, \chi_q > \tag{3.55}$$

$$= \begin{cases} 0 & \text{for } q < p \\ \frac{1}{2^q}\binom{q}{p} & \text{for } q \geq p \end{cases} \tag{3.56}$$

Arrays ${}^{l}R_{qp}$ and ${}^{r}R_{qp}$, $q, p = 0, \ldots, 5$ are presented in Fig. 8.

Constraints for 2-D Subparametric Elements

Since the shape functions for the 2-D master element are defined as tensor products of the 1-D functions, the results for the 1-D case hold exactly in the same form in the 2-D situation, the only difference being that the calculated constraint equations have to be applied to the proper degrees of freedom (see Fig. 9). It follows from the definition of the subparametric elements that the constraints coefficients are *exactly the same*, even when the elements have curved boundaries. This follows from the fact that the shape functions' behavior in a subparametric element on a part of its boundary depends exclusively upon the deformation of the part of the boundary, and therefore, any relation defined for the shape functions in the generic situation carries over immediately to the case of two small elements sharing an edge with a large element, so long as the deformation of the edge is identical in all three elements. The situation is illustrated in Fig. 9.

$$
{}^{l}R_{qp} = \begin{bmatrix}
1 & 1/2 & & & & & \\
 & 1/2 & & & & & \\
 & -1 & 1/4 & & & & \\
 & 0 & -3/8 & 1/8 & & & \\
 & -1 & 6/16 & -4/16 & 1/16 & & \\
 & 0 & -10/32 & 10/32 & -5/32 & 1/32 & \\
 & -1 & 15/64 & -20/64 & 15/64 & -6/64 & 1/64
\end{bmatrix}
$$

$$
{}^{r}R_{qp} = \begin{bmatrix}
1/2 & & & & & & \\
1/2 & 1 & & & & & \\
-1 & 1/4 & & & & & \\
0 & 3/8 & 1/8 & & & & \\
-1 & 6/16 & 4/16 & 1/16 & & & \\
0 & 10/32 & 10/32 & 5/32 & 1/32 & & \\
-1 & 15/64 & 20/64 & 15/64 & 6/64 & 1/64 &
\end{bmatrix}
$$

Figure 8: The constraint coefficients for a sixth order approximation. The unfilled coefficients are zero.

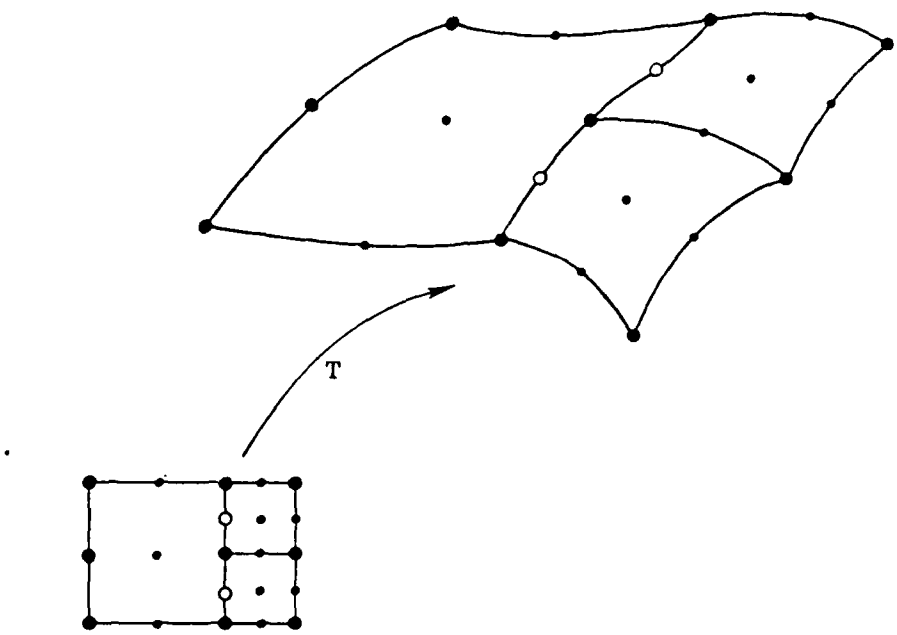


Figure 9: Illustration of the constraints for the subparametric elements

4 Some Details Concerning the Data Structure

In the classical FEM, elements as well as nodes are usually numbered consecutively in an attempt to produce a minimal band within the global stiffness matrix. When the program identifies an element to process its contribution to the global matrices, the minimal information needed is the node numbers associated with the element. Adaptive refinement and unrefinement algorithms require much more information on the mesh structure than the classical assembly process.

First of all, we introduce the notion of a *family*. Whenever an element is refined a new family is created. The original element is called the *father* of the family and the four new elements are called its *sons*. Graphically, the geneology on families can be presented in a family tree structure as illustrated in Fig. 10.

An examination of refinement and unrefinement algorithms (see [7] for details) reveals that for a given element NEL, one must have access the following information:

- element node numbers
- element neighbors
- the tree structure information, including: the
  - number of the element family
  - number of the father
  - numbers of sons
  - refinement level (number of generation)

For a given NODE we also require,

- node coordinates
- values of the degrees of freedom associated with the node

In general, some information is stored explicitly in a data base consisting of a number of arrays, some other information is recovered from the data base by means of simple algorithms. A careful balance should be maintained between the amount of information stored (storage requirements) and recovered (time).

The following is s short list of arrays used in the data base:

1. The tree structure is stored in the condensed, family-like fashion [6], [7] in two arrays

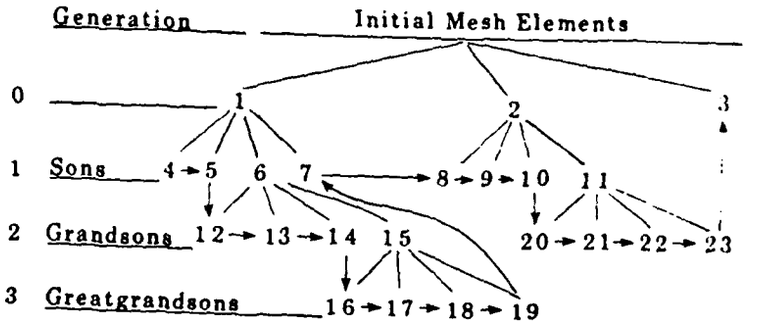


Figure 10: A tree structure and the natural order of elements: 4, 5, 12, 13, 14, 16, 17, 18, 19, 7, 8, 9, 10, 20, 21, 22, 23, 3.

NSON(NRELEI)

NTREE(5,MAXNRFAM)

where NRELEI is the number of elements in the initial mesh and MAXNRFAM is the anticipated maximum number of families. For an element NEL of the initial mesh, NSON(NEL) contains its first son number (if there is any). For a family NFAM, NTREE(1,NFAM) contains the number of the father of the family while the other four entries NTREE(2:5,NFAM) are reserved for the *"first-born"* sons of the sons of the family (the first-born "grandsons" of the father).

2. The initial mesh neighbor information is stored explicitly in array

   NEIG(4,NRELEI)

   containing up to four neighbors for each element of the initial mesh (elements adjacent to the boundary may have less neighbors).

3. For every *active* element, up to nine *nicknames* are stored in array

   NODES(9,MAXNRELEM)

   where MAXNRELEM is the anticipated maximum number of elements.

   For a regular node, the nickname is defined as

   NODE*100 + NORDER

   where NODE is the node number and NORDER the order of approximation associated with the node.

   For an irregular node, the nickname is defined as

   - NORDER

   where NORDER is again the order of approximation corresponding to the node.

4. For a particular component IEL of a vector-valued solution, the corresponding degrees of freedom are stored sequentially in array

   U(MAXNRDOF,IEL)

   where MAXNRDOF is the anticipated maximal number of degrees of freedom. Two extra integer arrays are introduced to handle the information stored in array U. Array

   NADRES(MAXNRNODE)

contains for every node, NODE, the address of the *first* from the degrees of freedom corresponding to NODE in array U. If K = NADRES(NODE) is such an address, the address for the next degree of freedom can be found in

NU(K)

and so on, until NU(K)=0, which means that the last degree of freedom for a node has been found (see [7] for a detailed discussion). The parameter MAXNRNODE above is the anticipated maximal number of nodes.

5. The node coordinates are stored in array

XNODE

XNODE(2,MAXNRNODE)

The rest of the necessary information is *reconstructed* from the data structure by means of simple algorithms. These include:

- calculation of up to eight neighbors for an element
- calculation of *local* coordinates of nine nodes for an element determining its geometry (the irregular nodes coordinates have to be reconstructed by interpolating regular nodes coordinates)
- recovery of the tree–structure related information, e.g., level of refinement, the sons numbers, etc.
- an algorithm establishing the *natural order of elements*

During the h and p refinements and unrefinements, both elements and nodes are created and deleted in a rather random way. This makes it impossible to denumerate them in a consecutive way, according to their numbers (for instance, as a result of unrefinements some numbers may be simply missing!). Thus a new ordering of elements has to be introduced which is based on some scheme other than an element numbers criterion. In the code discussed here, we use "the *natural order of elements*" based on the initial mesh elements ordering and the tree structure. The concept is illustrated in Fig. 10. One has to basically follow the tree of elements obeying the order of elements in the initial mesh and the order of sons in a family.

The natural order of elements may serve as a basis for defining an order for nodes and, consequently, for degrees of freedom, when necessary.

For a detailed discussion of the data structure as well as a critical review of different data structures in context of different h-refinement techniques, we refer again to [7].

4.1 References

5 References

1. Guo, B., and Babuška, I., "The *h-p* Version of the Finite Element Method," Parts 1 and 2, **Computational Mechanics** (1986) 1, 21–41, 203–220.

2. Babuška, I., and Suri, M., "The *h-p* Version of the Finite Element Method with Quasiuniform Meshes," *RAIRO Math. Mod. and Numer. Anal.*, 21, No. 2, 1987.

3. Gui, W., and Babuška, I., "The *h,p* and *h-p* Versions of the Finite Element Method in One Dimension, Parts 1, 2, 3," *Numer. Math.*, 49 (1986), 577–683.

4. Devloo, Ph., Oden, J. T., and Pattani, P., "An *h-p* Adaptive Finite Element Method for the Numerical Simulation of Compressible Flow," *Computer Methods in Applied Mechanics and Engineering*, Vol. 70, pp. 203–235, 1988.

5. Delves, L. M., and Hall, C., "An Implicit Matching Principle for Global Element Calculations" *J. Inst. of Mathematics and Its Applications* (1979), 23, 223–234.

6. Rheinboldt, W. C., and Mesztenyi, Ch. K., "On a Data Structure for Adaptive Finite Element Mesh Refinements," *ACM Transactions on Mathematical Software*, Vol. 6, No. 2, pp. 166–187, 1980.

7. Demkowicz, L., and Oden, J. T., "A Review of Local Mesh Refinement Techniques and Corresponding Data Structures in *h*-Type Adaptive Finite Element Methods," **TICOM Report 88-02**, The Texas Institute for Computational Mechanics, The University of Texas at Austin, Austin, Texas 78712.

8. Bank, R. E., Sherman, A. H., and Weiser, A., "Refinement Algorithms and Data Structures for Regular Mesh Refinement," in *Scientific Computing*, R. Stepleman, et al. (eds.), IMACS, North Holland, 1983.

9. Ciarlet, P. G., **The Finite Element Method for Elliptic Problems**, North Holland, Amsterdam, 1978.

10. Oden, J. T., and Carey, G. F., **Finite Elements: Mathematical Aspects**, Vol. IV, Prentice Hall, Englewood Cliffs, 1983.

APPENDIX B

Theory and Implementation of High–Order Adaptive *hp*-Methods for the Analysis of Incompressible Flows

1 Introduction

This appendix is presented as a summary of some approaches and some early numerical results on *h-p* methods for incompressible flows. The appendix is based on a forthcoming paper by J. T. Oden, with the same title as that above, that is to appear in **Computational Nonlinear Mechanics in Aerospace Engineering**, a volume in the "AIAA Progress in Aeronautics and Astronautics."

It is worth re-emphasizing the strategy behind the uses of *hp*-methodology. In important CFD circles, the inadequacy of contemporary computing devices for handling realistic three–dimensional flow simulations is frequently presented as justification for research on and development of larger and faster computations and concomitant larger budgets for computing. While few will argue against the need for bigger and better hardware, and we shall certainly not do that here, the premise for many of these widely published projections is rooted in the jargon of traditional, lower–order, finite difference approaches for structured meshes. A typical scenario is this:

Problem Class	*Number of Grid Points Required*	*Computer Class*
• 3D Steady Viscous Flow	$10^5 - 10^6$	VI
• Reynolds Averaged Navier-Stokes	10^7	$30 \times VI$
• Large Eddy Simulations	10^9	$3 \times 10^3 \times VI$
• Full Navier-Stokes	$10^{12} - 10^{15}$	$3 \times 10^9 \times VI$

Here "Class VI" refers to a sixth–generation supercomputer, such as a CRAY YMP, and "$3 \times 10^9 \times VI$" suggests that machines with a billion times the speed and storage of today's supercomputers will be needed to resolve flow features of interest in high Reynolds–number flows.

Of course, these figures are very rough and one can argue over the location of decimal points here or there, but they do indeed highlight the enormous challenge facing future advances of computational fluid dynamics. This challenge is real and is not at all at issue; what

is at issue is the notion that flow features must be resolved by merely adding and handling more gridpoints. This terminology suggests that the driving factor governing research on CFD is only the performance of the computing device itself and that the algorithms to be implemented on these devices will continue to be versions of the fixed gridpoint methods that have been popular in CFD for decades.

An alternative philosophy is to also look toward the development of new algorithms for CFD that are designed to streamline and optimize the computational process. These include "smart algorithms," which change in structure and performance during a flow calculation to accommodate changing properties of the solution. In particular, they include *adaptive finite element methods*, which are designed to adjust mesh parameters so as to control numerical error. Importantly, they include so–called h, p, and r-adaptive schemes which dynamically change the mesh size h, the local spectral order p, and the relocation r of gridpoints so as to deliver very high accuracies with near–minimal numbers of degrees-of-freedom.

Such adaptive methods need not function on structured meshes. Most importantly, they can, and generally do, produce *exponentially–convergent approximations*, meaning that such adaptive methods can produce results of a given accuracy using many orders-of-magnitude fewer degrees of freedom than that required by conventional approaches. Thus, flow features are not resolved by heuristically adding gridpoints where the analyst thinks they may be needed, but rather by automatically distributing element sizes and spectral orders in a way designed to produce results with a preset level of accuracy.

A remarkable by–product of these approaches is the calculation of estimates of the local (elementwise) error in the approximation. For example, to control the computational process, the calculated flowfield may be used to derive element residuals, and these residuals can be used to calculate estimates of error in appropriate norms. The quality and sophistication of existing estimation methods vary, but they have proven to be invaluable in giving the analyst an independent measure of the reliability of his calculation. Also, they provide a rational alternative to judging the convergence of a flow calculation by simply monitoring the change in results due to successive refinements.

While recognition of the significant potential of adaptive hp-methods is increasing in the CFD community, many research issues remain to be resolved. The present paper outlines a class of adaptive hp-methods for incompressible viscous flow problems and discusses results on the development of hp-strategies, error estimators, higher–order flow solvers, and results of representative numerical experiments. This work continues our study of adaptive techniques initiated in the early 1980s for both compressible and incompressible flow simulations.

Work on hp-adaptive methods in computational fluid dynamics began in 1983 and 1984 (see [1,2]) and originally focused on two–dimensional steady, viscous incompressible flow. Improvements in techniques for error estimation and in hp-data structures led to extensions

to compressible flow problems (e.g., [3–8]); for surveys on general adaptive finite element methods, see [9–11]. Smart algorithms and adaptive methods in CFD are discussed in [12–16]. An overview of h- and h-p finite element methods for elliptic problems is given in [17].

hp-finite element methods share features of standard fixed–order finite element methods and of spectral methods, both of which have a long history. For a history of finite element methods, see [18]. Global spectral methods, of course, data back to the nineteenth century and were used in a weak or variational setting by Ritz [19] in 1908, Galerkin [20] in 1915, and Kantorovich and Krylov [21] in 1936. The idea of using spectral approximations for finite element methods, with or without collocation, appeared very early in the finite element literature (see, e.g., [21]). The idea of using exact or high–order representations of functions over a regular subdomain and to then patch these subdomains together to produce a global model was the forerunner to "more practical" finite element methods based on low–order approximations and finer meshes. The problem was with interface conditions—how could these high–order spectral representations be effectively matched together?

The interface issue was rather trivially solved when only low–order conforming polynomial approximations were employed, and this fact led to the development of traditional and highly successful finite element schemes. The resurrection of high–order schemes within a finite element context came with the pioneering work of Szabo and his collaborators (e.g. [23–28]) on the p-version of the finite element method. There the notion of hierarchical families of polynomials on triangular meshes was developed, which simultaneously resolved the interface problem and led to techniques which produced remarkable accuracy for classes of elliptic problems in linear elasticity. The mathematical properties of p-version finite elements has been studied by Babuška, Szabo, and others [17,24,29–35] and is now a widely–accepted approach for solving problems in linear elasticity and in other areas. The spectral element methods developed by Patera and Maday and their colleagues (e.g., [36–40]) are, in fact, closely related to p-versions of the finite element technique. In this technique, Chebyschev polynomials and either collocation or Galerkin/Gauss-Lobbatto quadratures are used to produce high–order local spectral approximations on a fixed mesh. Interface conditions are enforced by point matching at boundary collocation points or by using a "weak" continuity condition in which interelement continuity is enforced by an L^2-projection of traces of functions on neighboring elements onto their common boundary.

A general approach to hp-adaptive methods was developed in a series of papers by Oden, Demkowicz, Rachowicz, and others, see [41–43]. There a general data structure, constructed on quadrilateral or hexahedral meshes, was developed which provided for irregular h refinements and p enrichments. An adaptive strategy is also given in these works together with a mesh optimization scheme designed to choose an acceptable sequence of h or p refinements to yield an optimal nonuniform hp mesh. An extension of several aspects of these works is

given in the present paper.

Following this Introduction, mathematical preliminaries, notations, and weak forms of the Navier-Stokes equations are given in Section 2. General properties of *hp*-finite element spaces are established in Section 3, and *hp*-finite element models of the Navier-Stokes equations are developed in Section 4. There we also discuss penalty approximations of the pressure and incompressibility constraint, which are used in developing higher-order flow solvers. Several high-order semi-implicit solvers are discussed in Section 5 and error estimation and adaptivity techniques are discussed in Section 6. Section 7 contains the results of several numerical experiments.

2 PRELIMINARIES

In this section, we record preliminary notations and problem definition for future reference.

2.1 The Governing Equations

We begin with the description of the flow of a viscous incompressible fluid embodied in the following problem:

Given an open bounded domain $\Omega \subset \mathbb{R}^n$, $n = 2$ *or* 3, *data* $(\boldsymbol{f}, \overline{\boldsymbol{u}}, \boldsymbol{g}, \boldsymbol{u}_0)$ *and time interval* $[0,T]$, *find a velocity field* $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x},t)$ *and a pressure* $P = P(\boldsymbol{x},t)$, $\boldsymbol{x} = (x_1, x_2, \cdots, x_n) \in \overline{\Omega}$, $t \in [0,T]$, *such that*

$$\left.\begin{aligned} \boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u} - 2\nu\nabla\cdot\boldsymbol{D}(\boldsymbol{u}) + \nabla P &= \boldsymbol{f} \\ \nabla\cdot\boldsymbol{u} &= 0 \end{aligned}\right\} \text{ in } \Omega\times(0,T] \tag{2.1}$$

$$\begin{aligned} &\boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{u}_0(\boldsymbol{x}) \quad \text{in } \overline{\Omega} \\ &\boldsymbol{u}(s,t) = \overline{\boldsymbol{u}}(s,t), \\ &\qquad s \in \Gamma_1,\ t \in [0,T] \\ &\boldsymbol{\sigma}(\boldsymbol{u},P)\cdot\boldsymbol{n}(s,t) = \boldsymbol{g}(s,t), \\ &\qquad s \in \Gamma_2,\ t \in [0,T] \end{aligned}$$

Here the following notations and conventions are used:

$\boldsymbol{u}_t$ = $\dfrac{\partial \boldsymbol{u}}{\partial t}$

$\boldsymbol{D}(\boldsymbol{u})$ = $\frac{1}{2}\left(\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^T\right)$ = the strain rate tensor

ν = μ/ρ = the kinematic viscousity, μ being the constant viscosity and ρ the density of the fluid

P = p/ρ = the kinematic pressure; p being the hydrostatic pressure

$\boldsymbol{f}$ = $\boldsymbol{f}(\boldsymbol{x},t) = \overline{\boldsymbol{f}}/\rho$ = the body force per unit volume, $\overline{\boldsymbol{f}}$ being the body force density

$\boldsymbol{u}_0$ = $\boldsymbol{u}_0(\boldsymbol{x})$ = the initial velocity distribution

$\overline{\boldsymbol{u}}$ = $\overline{\boldsymbol{u}}(s,t)$ = prescribed velocity on a portion Γ_1 of the boundary $\partial\Omega$ of Ω:
$\boldsymbol{x} \in \partial\Omega \Longrightarrow \boldsymbol{x} = \boldsymbol{x}(s)$, $\Gamma_1 \subset \partial\Omega$

$\boldsymbol{\sigma}(\boldsymbol{u},P)$ = the (density–weighted) Cauchy stress tensor = $\rho(2\nu\boldsymbol{D}(\boldsymbol{u}) - P\mathbf{1})$, $\mathbf{1}$ being the unit tensor

$\boldsymbol{n}$ = a unit exterior normal to $\partial\Omega$

$\boldsymbol{g}$ = $\boldsymbol{g}(s,t)$ = prescribed traction on $\Gamma_2 \subset \partial\Omega$,
$\partial\Omega = \overline{\Gamma_1 \cup \Gamma_2}$, $\Gamma_1 \cap \Gamma_2 = \emptyset$

Remarks:

2.1) The boundary and initial data must satisfy certain compatibility conditions if (2.1) is to represent a well–posed problem. In particular, we must have

$$\boldsymbol{\nabla} \cdot \boldsymbol{u}_0 = 0 \text{ and } \boldsymbol{u}_0 = \overline{\boldsymbol{u}}(s,0) \text{ on } \Gamma_1$$

$$\int_{\partial\Omega} \boldsymbol{u}_0 \cdot \boldsymbol{n}\, ds = 0$$

We demand that meas $\Gamma_i > 0$ or $\Gamma_i = \emptyset$, $i = 1,2$, then, if $\Gamma_2 = \emptyset$, $\overline{\boldsymbol{u}}$ must satisfy,

$$\int_{\Gamma_1 = \partial\Omega} \overline{\boldsymbol{u}} \cdot \boldsymbol{n}\, ds = 0 \qquad t \in [0,T]$$

2.2) Boundary conditions other than those given in (2.1) can be incorporated into the analysis. See, e.g., [44]. ∎

2.2 Weak Formulations

To recast (2.1) into a functional setting, we introduce the following notations:

Spaces

$$\left.\begin{aligned} V &= \left\{\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}) \in (H^1(\Omega))^n \,|\gamma_0 \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_1\right\} \\ H &= \{\boldsymbol{v} \in V | div\ \boldsymbol{v} = 0 \text{ in } \Omega\} \\ M &= \left\{q \in L^2(\Omega) | \int_\Omega q\ dx = 0 \text{ if } \Gamma_2 = \emptyset\right\} \end{aligned}\right\} \tag{2.2}$$

When the domain of functions needs emphasis, we will also use the notations $V = V(\Omega)$, $H = H(\Omega)$, and $M = M(\Omega)$. We also introduce the norms,

$$\begin{aligned} \|\boldsymbol{v}\|_{1,\Omega}^2 &= \int_\Omega (\boldsymbol{\nabla v} : \boldsymbol{\nabla v} + \boldsymbol{v} \cdot \boldsymbol{v}) dx, \\ \|q\|_{0,\Omega}^2 &= \int_\Omega q^2 dx,\ \|\boldsymbol{v}\|_{r,\Omega}^2 = \sum_{i=1}^n \|v_i\|_{H^r(\Omega)}^2,\ r > 1,\ \text{etc.} \end{aligned}$$

Forms

$$\left.\begin{aligned} &a : V \times V \to \mathbb{R} \\ &a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega 2\nu \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v})\ dx \\ &b : V \times M \to \mathbb{R} \\ &b(\boldsymbol{v}, q) = \int_\Omega q\ div\ \boldsymbol{v}\ dx \\ &c : V \times V \times V \to \mathbb{R} \\ &c(\boldsymbol{u}, \boldsymbol{v}\ ;\ \boldsymbol{w}) = \int_\Omega (\boldsymbol{u} \cdot \boldsymbol{\nabla v}) \cdot \boldsymbol{w}\ dx \\ &L : V \to \mathbb{R} \\ &\qquad L(\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}) + \int_{\Gamma_2} \boldsymbol{g} \cdot \boldsymbol{v}\ ds \end{aligned}\right\} \tag{2.3}$$

Here $H^1(\Omega)$ is the usual Sobolev space of functions with L^2-distributional derivatives defined on Ω, γ_0 is the trace map of $(H^1(\Omega))^n$-functions onto $\partial\Omega$ ($\gamma_0 \boldsymbol{v} = \boldsymbol{v}|_{\partial\Omega}$, $\boldsymbol{v} \in \left(C^0\left(\overline{\Omega}\right)\right)^n$, and we shall simply write $\gamma_0 \boldsymbol{v} = \boldsymbol{v}$ unless confusion is likely), and $(\boldsymbol{f}, \boldsymbol{v}) = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v}\ dx$, $dx = dx_1 dx_2 \cdots dx_n$. Here also the notations $\boldsymbol{A} : \boldsymbol{B} = \sum_{i,j=1}^n A_{ij} B_{ij}$ and $\boldsymbol{a} \cdot \boldsymbol{A} \cdot \boldsymbol{b} = \sum_{i,j=1}^n a_i A_{ij} b_j$ are used for tensors $\boldsymbol{A}, \boldsymbol{B}$ and vectors $\boldsymbol{a}, \boldsymbol{b}$. A weak formulation of (2.1) can now be presented as follows:

Given $\boldsymbol{f} \in L^2(0,T;L^2(\Omega)), \boldsymbol{g} \in L^2(0,T;L^2(\Gamma_2)), \boldsymbol{u}_0 \in H$, *and* $\hat{\boldsymbol{u}} \in (H^1(\Omega))^n$, $\hat{\boldsymbol{u}}|_{\Gamma_1} = \overline{\boldsymbol{u}}$, *find* $\boldsymbol{u} = \hat{\boldsymbol{u}} + L^2(0,T;V)$ *and* $P \in L^2(0,T;M)$, $\boldsymbol{u}(0) = \boldsymbol{u}_0$, *such that*

$$\begin{aligned} (\boldsymbol{u}_t, \boldsymbol{v}) + c(\boldsymbol{u}, \boldsymbol{u}; \boldsymbol{v}) + a(\boldsymbol{u}, \boldsymbol{v}) - b(\boldsymbol{v}, P) &= L(\boldsymbol{v}) \\ b(\boldsymbol{u}, q) &= 0 \\ \forall\, (\boldsymbol{v}, q) \in V &\times M \end{aligned} \tag{2.4}$$

Conditions on the regularity of the data $(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{u}_0, \overline{\boldsymbol{u}})$ can, of course, be weakened; see Temam [45].

The Stokes problem arises as a special case of (2.4) when $\boldsymbol{u}_t = \boldsymbol{0}$ and the convection term $c(\cdot\ ,\ \cdot\ ;\ \cdot)$ is dropped:

Find

$(\boldsymbol{u}, P) \in V \times M$:

$$\begin{aligned} a(\boldsymbol{u}, \boldsymbol{v}) - b(\boldsymbol{v}, P) &= L(\boldsymbol{v}) \\ b(\boldsymbol{u}, q) &= 0 \\ \forall\, (\boldsymbol{v}, q) &\in V \times M \end{aligned} \tag{2.5}$$

2.3 Weak Forms on Partitions of Ω

It is convenient to also define weak formulations of the Navier-Stokes equations on partitions of Ω. Let $\mathcal{Q}_h$ denote a family of partitions $\mathcal{T}_h$ of Ω into subdomains K:

$$\overline{\Omega} = \bigcup_{h>0} \{K \in \mathcal{T}_h\}, \quad K \bigcap J = \Gamma_{KJ} \quad \forall\, K, J \in \mathcal{T}_h$$

$$h_K = dia\,(K), \quad h = \sup_{K \in \mathcal{T}_h} h_K, \quad \mathcal{T}_h \in \mathcal{Q}_h$$

We denote by $V(K)$, $H(K)$, and $M(K)$ the spaces of restrictions of functions in V, H, and M, respectively, to subdomain K. Let $a_K(\cdot,\cdot)$, $b_K(\cdot,\cdot)$, $c_K(\cdot,\cdot\ ;\ \cdot)$, and $L_K(\cdot)$ denote forms analogous to (2.3) defined on these restricted spaces; i.e.,

$$\left.\begin{aligned} a_K(\boldsymbol{u},\boldsymbol{v}) &= \int_K 2\nu \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v})\, dx \\ b_K(\boldsymbol{v},q) &= \int_K q\ div\ \boldsymbol{v}\ dx \\ c_K(\boldsymbol{u},\boldsymbol{v};\boldsymbol{w}) &= \int_K (\boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{v})\cdot\boldsymbol{w}\ dx \\ L_K(\boldsymbol{v}) &= (\boldsymbol{f},\boldsymbol{v})_K + \int_{\Gamma_2\cap\partial K} \boldsymbol{g}\cdot\boldsymbol{v}\ ds \end{aligned}\right\} \tag{2.6}$$

with $(\boldsymbol{f},\boldsymbol{v})_K = \int_K \boldsymbol{f}\cdot\boldsymbol{v}\ dx$. Then $a(\boldsymbol{u},\boldsymbol{v}) = \sum_{K\in\mathcal{T}_h} a_K(\boldsymbol{u},\boldsymbol{v})$, $b\,(\boldsymbol{v},q) = \sum_{K\in\mathcal{T}_h} b_K(\boldsymbol{v},q)$, ..., etc., and the weak form of the Navier-Stokes equations for a typical subdomain K assumes the form,

Find $\boldsymbol{u}_K \in L^2(0,T;V(K))$, $P_K \in L^2(0,T;M(K))$ *such that*

$$\begin{aligned} (\boldsymbol{u}_{Kt},\boldsymbol{v})_K \ &+\ c_K\,(\boldsymbol{u}_K,\boldsymbol{u}_K;\boldsymbol{v}) + a_K\,(\boldsymbol{u}_K,\boldsymbol{v}) \\ -\,b_K\,(\boldsymbol{v},P_K) \ &=\ L_K(\boldsymbol{v}) + \int_{\partial K\setminus\Gamma_1} \boldsymbol{v}\cdot\boldsymbol{\sigma}\,(\boldsymbol{u}_K,P_K)\cdot\boldsymbol{n}_K ds \\ b(\boldsymbol{u}_K,q) \ &=\ 0 \\ &\qquad \forall\ \boldsymbol{v}\in V(K),\ \forall\ q\in M(K) \\ \boldsymbol{u}_K(0) \ &=\ \boldsymbol{u}_0|_K(0)\ ,\ \boldsymbol{u}_K = \overline{\boldsymbol{u}}\ \textit{on}\ \Gamma_1\cap\partial K \end{aligned} \tag{2.7}$$

for given data $(\boldsymbol{f},\boldsymbol{g},\boldsymbol{u}_0,\overline{\boldsymbol{u}})$, $\boldsymbol{n}_K$ *being the unit exterior normal to* ∂K.

We construct approximations of (2.4) by using *hp*-approximations of the spaces V and M in Section 4.

3 *hp* FINITE ELEMENT APPROXIMATIONS

We now take up the fundamental issue of constructing *hp*-finite element approximations of the spaces V, H, and M and $V(K)$, $H(K)$, and $M(K)$.

3.1 Shape Functions

To simplify the discussion, we begin with the two-dimensional case illustrated in Fig. 1. There we see the domain Ω partitioned into a mesh of elements K each of which is the image of a master element $\widehat{K}$ under a smooth invertible map F_K:

$$\boldsymbol{x} = (x_1, x_2) = F_K(\boldsymbol{\xi}) \ , \quad \boldsymbol{\xi} = (\xi_1, \xi_2)$$

Let

$$\widehat{K} = [-1,1] \times [-1,1]$$

We next record properties of special shape functions on $\widehat{K}$.

Lemma 3.1. Let $\xi \in [-1,1]$ *and denote*

$$\begin{aligned} \varphi_j(\xi) &= \left(\frac{2j-1}{2}\right)^{\frac{1}{2}} \int_0^\xi P_{j-1}(s)ds \qquad j \geq 2 \\ &= \frac{1}{\sqrt{2\,(2j-1)}} \left(P_j(\xi) - P_{j-2}(\xi)\right) \end{aligned} \tag{3.1}$$

where $P_j(\xi)$ *is the Legendre polynomial of degree* j. *Then*

$$\left.\begin{array}{l} \varphi_j \in \mathcal{P}_j([-1,1]) = \textit{the space of polynomials of degree } j \textit{ on } [-1,1] \\ \varphi_j(\pm 1) = 0 \\ (\varphi_j, \varphi_k) = \left\{ \begin{array}{ll} \left[(2j+1)^{-1} + (2j-3)^{-1}\right]/2(2j-1) & \textit{if } \ j = k \\ -(2j+1)^{-1}/2\left[(2j-1)(2j+2)\right]^{\frac{1}{2}} & \textit{if } \ j = k-2 \\ -(2j-3)^{-1}/2\left[(2j-1)(2j-5)\right]^{\frac{1}{2}} & \textit{if } \ j = k+2 \\ 0 & \textit{if otherwise} \end{array} \right. \\ \left(\varphi_j', \varphi_k'\right) = \left\{ \begin{array}{ll} \left[(2j-1)/2\right]/(2j-1) & \textit{if } \ j = k \\ 0 & \textit{if otherwise} \end{array} \right. \end{array}\right\} \tag{3.2}$$

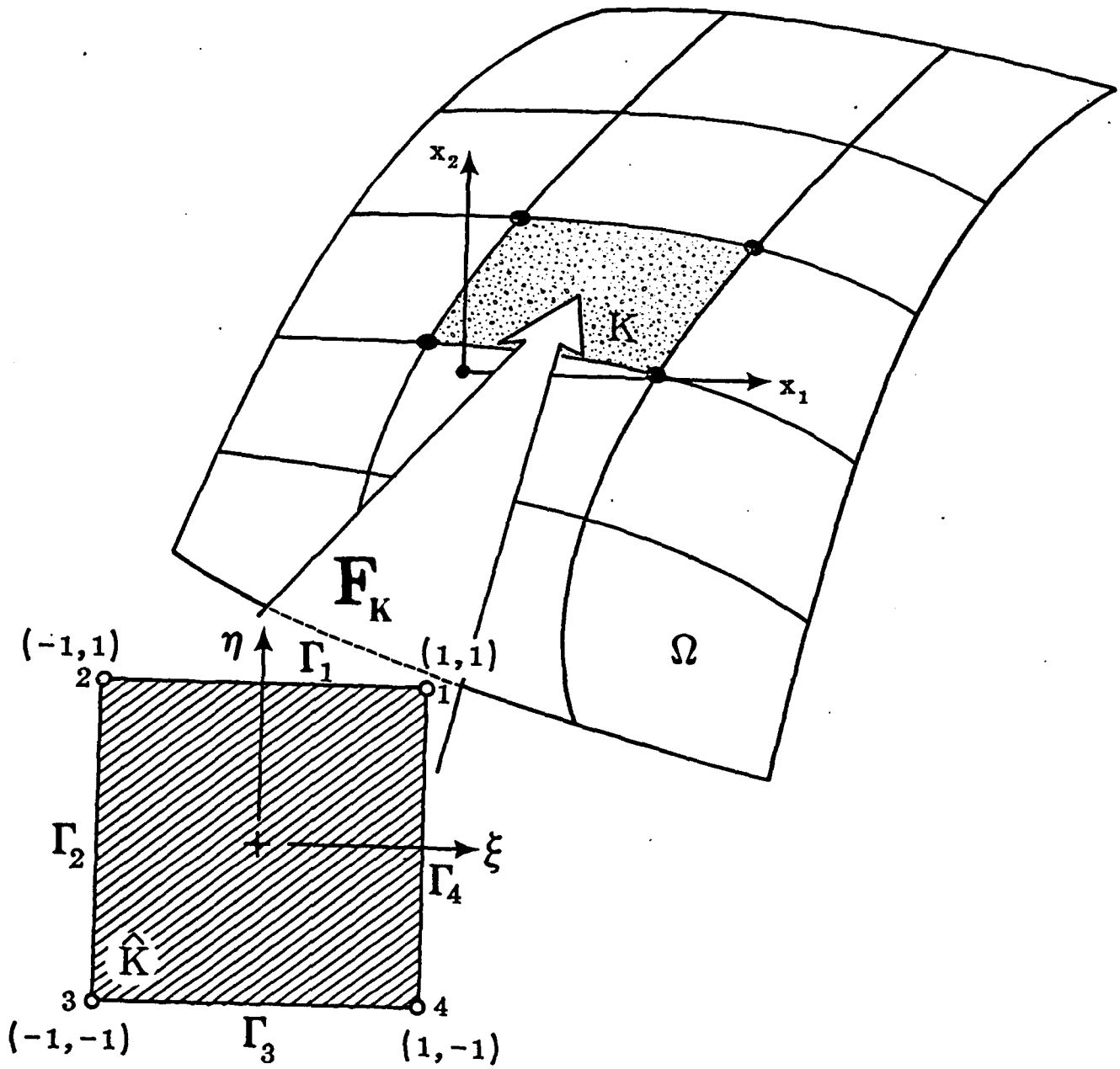


Figure 1: Master element $\widehat{K}$ and bijective map F_K of $\widehat{K}$ onto a partition of Ω into elements K.

where $(\varphi, \psi) = \int_{-1}^{1} \varphi\psi d\xi$.

Proof: These results follow from well-known properties of Legendre polynomials. See [46] and, particularly, [47]. ∎

With these notations in hand, we define the master element shape functions in three groups:

Node Functions. These are the standard bilinear shape functions

$$\chi^{\Delta}(\xi,\eta) = \frac{1}{4}(1 \pm \xi)(1 \pm \eta) \ , \ \Delta = 1,2,3,4 \tag{3.3}$$

$(\xi = \xi_1, \eta = \xi_2)$

Side Functions. On each side Γ_Δ of the boundary $\partial\widehat{K}$ of $\widehat{K}$, we define

$$\begin{aligned} \chi^{1,k}_{\partial\widehat{K}}(\xi,\eta) &= \frac{1}{2}(1+\eta)\varphi_k(\xi) \\ \chi^{2,k}_{\partial\widehat{K}}(\xi,\eta) &= \frac{1}{2}(1-\xi)\varphi_k(\eta) \\ \chi^{3,k}_{\partial\widehat{K}}(\xi,\eta) &= \frac{1}{2}(1-\eta)\varphi_k(\xi) \\ \chi^{4,k}_{\partial\widehat{K}}(\xi,\eta) &= \frac{1}{2}(1+\xi)\varphi_k(\eta) \end{aligned} \tag{3.4}$$

$$k = 2,3,\cdots,p$$

Bubble Functions. Internally, we define functions which vanish on $\partial\widehat{K}$ by

$$\chi^{B}_{jk}(\xi,\eta) = \varphi_j(\xi)\varphi_k(\eta) \ , \ 2 \le j,k \le p \tag{3.5}$$

The space $Q^p\left(\widehat{K}\right)$ of master element shape functions now defined by

$$Q^p\left(\widehat{K}\right) = \text{span}\left\{\chi^{\Delta}, \chi^{\Gamma,k}_{\partial\widehat{K}}, \chi^{B}_{jk} : 1 \le \Gamma,\Delta \le 4, \ 2 \le j,k \le p\right\} \tag{3.6}$$

In three dimensions, similar choices of shape functions are used except that there it is convenient to decompose the set into edge and face functions as well (see Fig. 2). For instance, for $\widehat{K} = [-1,1]^3$, $\xi = \xi_1$, $\eta = \xi_2$, $\zeta = \xi_3$, we have

3D Node Functions

$$\chi^{\Delta}(\xi,\eta,\zeta) = \frac{1}{8}(1 \pm \xi)(1 \pm \eta)(1 \pm \zeta)\ ,\ \Delta = 1,2,\cdots,8 \tag{3.7}$$

Edge Functions

$$\begin{aligned} \chi^{k}_{1,\partial\widehat{K}}(\xi,\eta,\zeta) &= \frac{1}{4}(1+\xi)(1+\zeta)\varphi_k(\eta) \\ \chi^{k}_{2,\partial\widehat{K}}(\xi,\eta,\zeta) &= \frac{1}{4}(1+\xi)(1-\zeta)\varphi_k(\eta) \\ &\vdots \\ \chi^{k}_{12,\partial\widehat{K}}(\xi,\eta,\zeta) &= \frac{1}{4}(1-\xi)(1-\eta)\varphi_k(\zeta) \\ & \qquad k = 2,3,\cdots,p \end{aligned} \tag{3.8}$$

Face Functions

$$\begin{aligned} \chi^{1}_{jk}(\xi,\eta,\zeta) &= \frac{1}{2}(1+\xi)\chi^{B}_{jk}(\eta,\zeta) \\ \chi^{2}_{jk}(\xi,\eta,\zeta) &= \frac{1}{2}(1-\xi)\chi^{B}_{jk}(\eta,\zeta) \\ &\vdots \\ \chi^{6}_{jk}(\xi,\eta,\zeta) &= \frac{1}{2}(1-\zeta)\chi^{B}_{jk}(\xi,\eta) \\ & \qquad 2 \le j,k \le p \end{aligned} \tag{3.9}$$

Bubble Functions

$$\begin{aligned} \chi^{B}_{jk\ell}(\xi,\eta,\zeta) &= \varphi_j(\xi)\varphi_k(\eta)\varphi_\ell(\zeta) \\ & \qquad 2 \le j,k,\ell \le p \end{aligned} \tag{3.10}$$

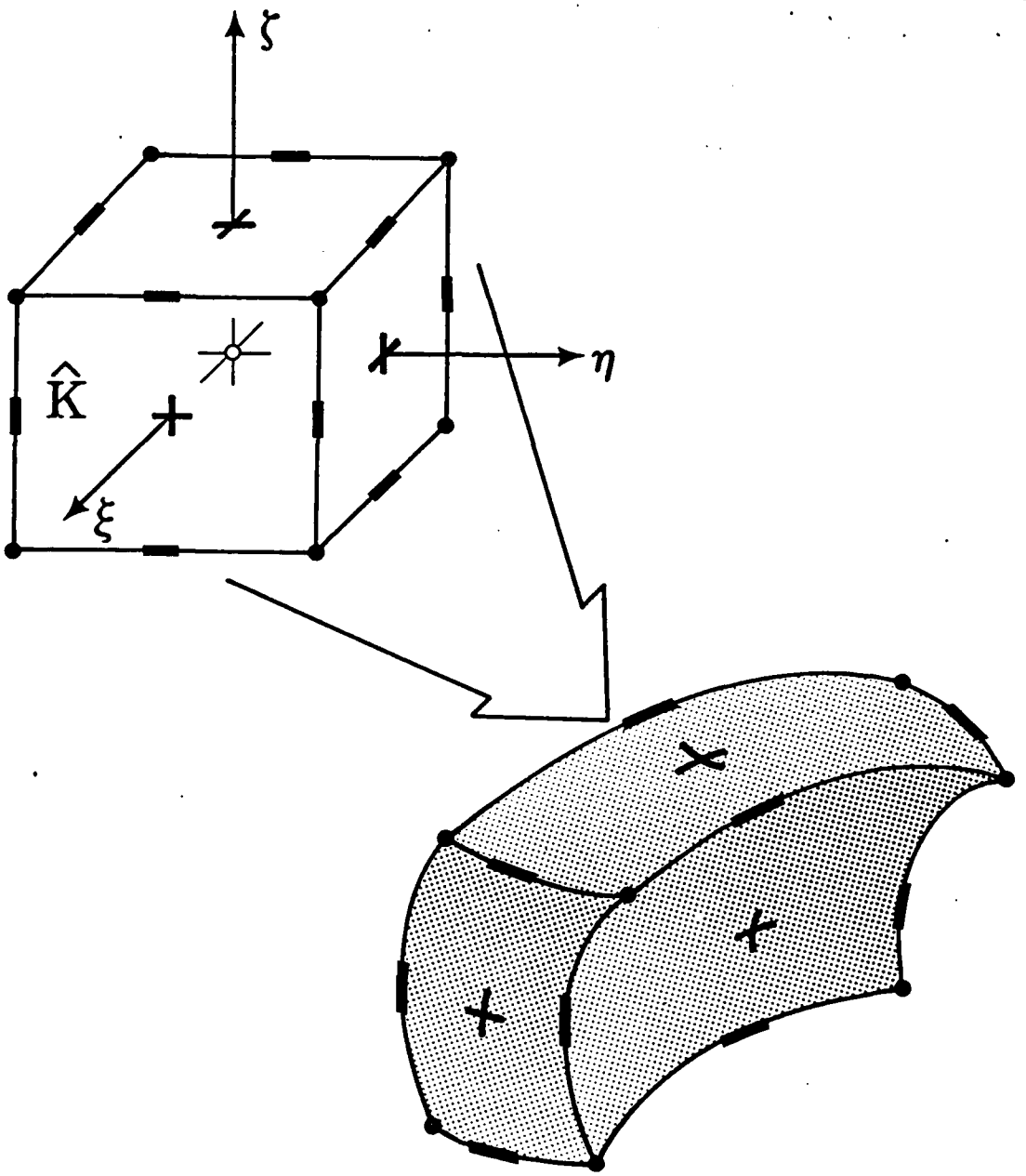


Figure 2: Nodal, edge, facial, and interior (bubble function) degree-of-freedom locations on a master element in three dimensions and a parametric image of the element in a curvilinear mesh.

Then we denote the space of master element shape functions by

$$Q^p\left(\widehat{K}\right) = \text{span}\left\{\chi^{\Delta}, \chi^k_{E,\partial\widehat{K}}, \chi^F_{jk}, \chi^B_{jk\ell} : 1 \le \Delta \le 8, 1 \le E \le 12, 1 \le F \le 6, 2 \le j,k,\ell \le p\right\} \tag{3.11}$$

Denote dim $Q^p(\widehat{K}) = \widehat{N} = (p+1)^2$. Then each $\widehat{u} \in Q^p(\widehat{K})$ representable in the form

$$\widehat{u}(\boldsymbol{\xi}) = \sum_{i=1}^{\widehat{N}} a^i \widehat{\psi}_i(\boldsymbol{\xi})$$

with $\widehat{\psi}_i$ the (relabeled) shape functions defining the basis of $Q^p(\widehat{K})$.

3.2 Subparametric Maps

Returning to the case (3.3) for convenience, we choose for the element maps F_K the subparametric functions,

$$\begin{aligned} \boldsymbol{x} = F_K(\boldsymbol{\xi}) &= F_K(\xi,\eta) \\ &= \boldsymbol{a}_K + \sum_{\Delta=1}^{4} \boldsymbol{x}_\Delta \chi^{\Delta}(\xi,\eta) + \sum_{\Gamma=1}^{4}\sum_{k=2}^{S} \boldsymbol{x}^k_\Delta \chi^{\Gamma,k}_{\partial\widehat{K}}(\xi,\eta) \end{aligned} \tag{3.12}$$

where $\boldsymbol{a}_K$ is a constant translation vector, $\boldsymbol{x}_\Delta = (x^\Delta_1, x^\Delta_2)$ are the coordinates of node (vertex) Δ of element K, and $\boldsymbol{x}^k_\Delta = (x^\Delta_{1k}, x^\Delta_{2k})$ are the coordinates of points interior to the edges Γ^Δ of ∂K. Often we take only $S = 2$ so that element geometry is modeled using subparametric quadratic approximation of curvilinear boundaries.

3.3 *h-p* Meshes and Data Structure

The element shape functions described above are hierarchical, meaning that the parameter p can be increased or decreased with the result that the associated element matrices for $p = p_1$ are always submatrices for $p = p_2$, $p_2 \ge p_1$. We also treat the mesh size h ($h_K = dia(K)$) as a parameter by employing standard h-adaptive techniques for bisection of elements discussed, for example, in [4]. Conformity of shape functions across interelement boundaries is accomplished using constraint approximations of the type introduced in [41]. The basic ideas are summarized as follows:

1. Let $\mathcal{T}_h \in \mathcal{Q}_h$ be a partition of Ω into $M = M(h)$ elements $K, \overline{\Omega} = \cup\{K \in \mathcal{T}_h\}$. Let $S^{hp}(K)$ denote the space of local finite element functions defined over element K,

$$S^{hp}(K) = \Big\{ \quad v = v(\boldsymbol{x}) = \sum_{i=1}^{N} \beta_i \psi_i(\boldsymbol{x}) |\ K \in \mathcal{T}_h,$$

$$\boldsymbol{x} \in K\ ,\ \psi_i = \widehat{\psi}_i \circ F_K^{-1}\ ,\ \widehat{\psi}_i \in Q^p\left(\widehat{K}\right),$$

$$\beta_i \in \mathbb{R} \quad \Big\}$$

where $Q^p(\widehat{K})$ is defined in (3.6).

2. Let $\Sigma(K) = \left(S^{hp}(K)\right)'$ denote the space of degrees-of-freedom corresponding to $S^{hp}(K)$; i.e., $\Sigma(K)$ can be identified as the dual space of $S^{hp}(K)$ so that $\Sigma(K)$ contains linear functionals q_K^i. In particular, the functionals q_K^i can be chosen to be a dual basis to the basis functions $\psi_i \in S^{hp}(K)$:

$$\left\langle q_K^i, \psi_j \right\rangle = \delta_j^i\ ,\ 1 \le i, j \le N$$

where $\langle \cdot, \cdot \rangle$ denotes duality pairing on $\Sigma(K) \times S^{hp}(K)$. Then, if

$$v \in S^{hp}(K)$$

$$v(\boldsymbol{x}) = \sum_{i=1}^{N} v^i \psi_i(\boldsymbol{x})$$

we must have

$$v^i = q_K^i(v) = \left\langle q_K^i, v \right\rangle$$

The functionals q_K^i (the degrees-of-freedom) consist of nodal values,

$$q_K^{\Delta}(u) = u\left(\boldsymbol{x}_K^{\Delta}\right) \qquad \Delta = 1, 2, 3, 4$$

and directional derivatives at the sides Γ^{Δ} or mixed derivatives at the centroid of K; e.g.,

$$q\ :\ u \longrightarrow D_{\boldsymbol{x}}^k u\left(\boldsymbol{x}_K^{\Delta}\right) \cdot (\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_k)\ ,\ k = 0, 1, \cdots$$

where $D_{\boldsymbol{x}}^k u$ is the k-th order differential of u.

3. The space $\widetilde{X}(\Omega)$ of *unconstrained* degrees of freedom consists of the set of functions u_h: $\Omega \to \mathbb{R}$ such that

   i) $u_h|_K \in S^{hp}(K) \qquad \forall\ K \in \mathcal{T}_h$

and

ii) for every pair of elements $K, J \in \mathcal{T}_h$ sharing a node $\boldsymbol{x}^\Delta \in K \cap J$, and every pair of degrees of freedom,

$$q_K \in \Sigma(K),\ q_J \in \Sigma(J) \text{ with}$$

$$q_K(u) = q_J(u) = D^k_{\boldsymbol{x}} u\left(\boldsymbol{x}^\Delta\right) \cdot (\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \cdots, \boldsymbol{\xi}_k)$$

$k = 0, 1, \cdots$, we have

$$q_K(u|_K) = q_J(u|_J)$$

4. The *global* degrees of freedom $\Sigma(\Omega)$ are linear functionals on $\widetilde{X}(\Omega)$:

$$\Sigma : \widetilde{X}(\Omega) \to \mathbb{R}\,, \qquad \Sigma(u) = q_K(u|_K)$$

and the (*discontinuous*) global base functions $\tilde{\chi}_k$ are defined so that

$$\left\langle \Sigma^j, \tilde{\chi}_k \right\rangle = \delta^j_k\,,\ 1 \le j, k \le \widetilde{N}$$

$\widetilde{N} = \dim \widetilde{X}(\Omega)$.

5. Now a conforming set of finite element basis functions χ are obtained by imposing continuity constraints. For this purpose, we partition the degrees of freedom into sets N^a and N^c of active and constrained (indices of) degrees of freedom, respectively. Then, for each constrained degree of freedom Σ^i, $i \in N^c$, there is a set $I(i) \subset N^a$ such that (see [41])

$$\Sigma^i(u_h) = \sum_{j \in I(i)} R^i_j \Sigma^j(u_h)$$

where R^i_j defines the Bookean map of active into constrained sets. For $\widehat{K}$, the entries R^i_j are rational numbers the exact nature of which depends upon the h-adaptive strategy. The final global finite element space is then

$$\left.\begin{aligned} X^{hp}(\Omega) &= \left\{ v_h \in \widetilde{X}^h(\Omega) | \Sigma^i(v_h) = \sum_{j \in I(i)} R^i_j \Sigma^j(v_h),\ \forall\, i \in N^c \right\} \\ &= \text{span}\,\{\chi_i\} \\ \chi_i(\boldsymbol{x}) &= \tilde{\chi}_i(\boldsymbol{x}) + \sum_{j \in S(i)} R^j_i \tilde{\chi}_j\,,\ i \in N^a \\ S(i) &= \{ j \in N^c | i \in I(j) \} \end{aligned}\right\} \tag{3.13}$$

3.4 The *hp*-Finite Element Spaces

The construction outlined in the previous subsection leads to the definition of families of finite–dimensional spaces of functions of the type (for the two–dimensional case)

$$\left.\begin{aligned} & X^{hp}(\Omega) \subset H^1(\Omega)\ ,\ \overline{\Omega} = \bigcup \{K \in \mathcal{T}_h\} \\ & X^{hp}(\Omega) = \left\{ v \in C^0(\Omega) \mid v|_K \in S^{hp}(K) \bigoplus_{k=1}^{4} E_K^k(\partial K) \right\} \\ & M^{hp}(\Omega) = \left\{ q \in L^2(\Omega) \mid q|_K \in S^{hp}(K) \bigoplus_{k=1}^{4} E_K^k(\partial K) \right\} \end{aligned}\right\} \tag{3.14}$$

We use the notations:

p_K = *the order of the highest–degree complete polynomial represented in the shape functions defined on element* K

$E_K^k(\partial K)$ = *the spaces of functions of the type* (3.4) *defined on the edges* Γ_K^k *of* K *which are introduced to fulfill continuity requirements across element boundaries*

Thus, for a typical constrained element K in the mesh, such as is shown in Fig. 3, continuity of the global basis functions is maintained by *enriching* the edge functions of K to match the highest order polynomial experienced on the common interelement boundary. When adjoining element edge functions are polynomials of different degree, the edge functions on common boundaries are enriched by the addition of edge functions of a degree equal to the largest polynomial degree of any element sharing that edge. Thus, for the *hp*-mesh shown in Fig. 4, shape functions of a given degree may overlap those of adjacent elements which support lower degree shape functions. This is illustrated in the color–coded figure by colored patches extending over common element edges. In three–dimensional *hp*-meshes, the implementation of this strategy requires also the addition of element edge functions as well. Once continuity is established, the highest order of complete polynomials contained in the element is denoted p_K.

Typical interpolation properties of *hp*-spaces are indicated in the following:

Proposition 3.1. *Consider a mesh* $\mathcal{T}_h$ *of elements* K *which are affine equivalent to a master element* $\widehat{K}$*, with*

$$\widehat{K} = [-1,1]^2\ ,\ \overline{\Omega} = \bigcup \{K \in \mathcal{T}_h\}\ ,\ F_K : \widehat{K} \to K$$
$$K \ni \boldsymbol{x} = \boldsymbol{T}_K \hat{\boldsymbol{x}} + \boldsymbol{a}_K\ , \tag{3.15}$$

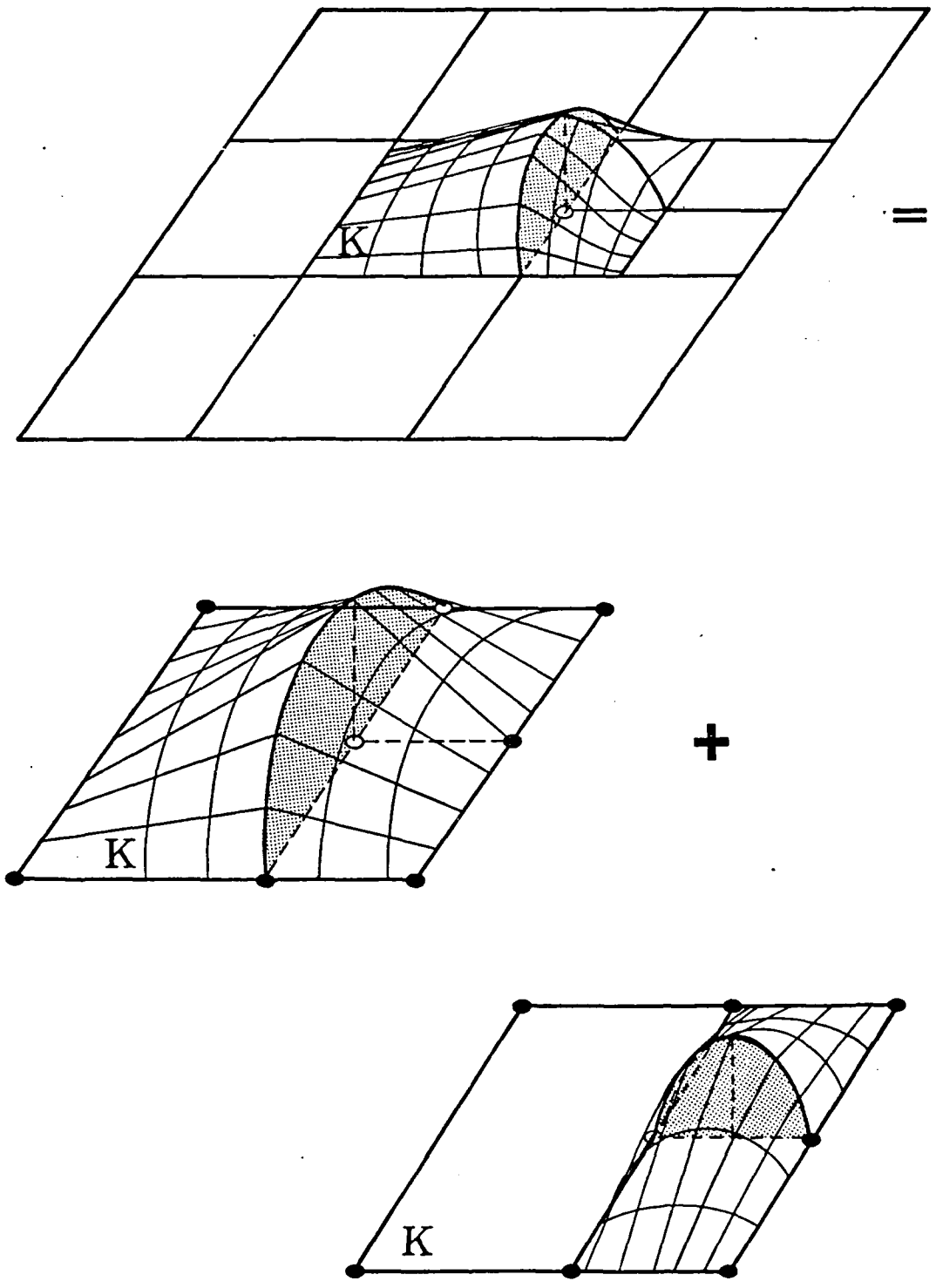


Figure 3: Illustration of the addition of edge polynomials at constrained edges so as to enforce continuity of the basis functions across interelement boundaries.

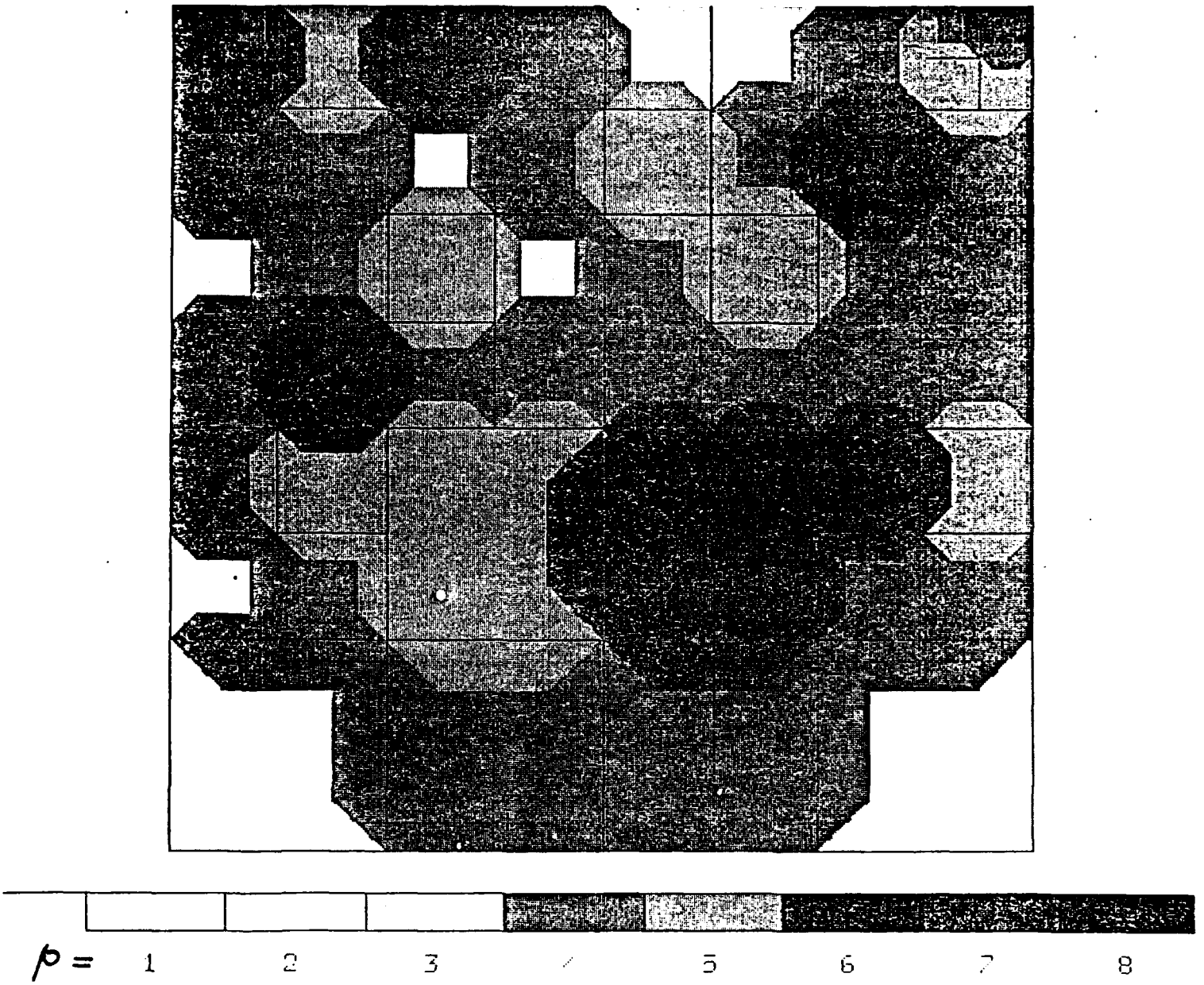


Figure 4: Color-coded nonuniform hp-mesh.

T_K being an invertible (constant) matrix and a_K a translation vector. Let

$$\rho_K = sup\{dia\ S \mid S = sphere\ contained\ in\ K\}$$

$$h_K = dia\ (K)$$

and suppose

$$\frac{h_K}{\rho_K} \leq \sigma = const. \qquad \forall\ K \in \mathcal{T}_h$$

Then, given any function $u \in H^r(K)$, there exists a sequence of interpolants $w^{hp} \in \mathcal{P}_{p_K}(K)$, the space of polynomials of degree $\leq p_K$ defined on K, $p_K = 1, 2, \cdots$, and a constant C, independent of u, p_K, or h_K, such that for any s, $0 \leq s \leq r$,

$$\left\| u - w^{hp} \right\|_{s,K} \leq C \frac{h_K^{\mu - s}}{p_K^{r-s}} \left\| u \right\|_{r,K} \tag{3.16}$$

where

$$\mu = \min\left(p_K + 1, r\right) \tag{3.17}$$

Proof: See, for example, [17]. ∎

Returning to (2.1), we assume that Ω is such that families of spaces $V^{hp}(\Omega)$, $M^{hs}(\Omega)$ can be constructed such that

1. $$\left.\begin{array}{rcl} V^{hp}(\Omega) = \left(X^{hp}(\Omega)\right)^n & \subset & V(\Omega) \\ M^{hs}(\Omega) & \subset & M(\Omega) \end{array}\right\} \tag{3.18}$$

2. $\forall\ \boldsymbol{v} \in V(\Omega) \bigcap (H^r(\Omega))^n$, $\exists\ \boldsymbol{w}^{hp} \in V^{hp}(\Omega)$

   such that

   $$\left\| \boldsymbol{v}\,\right|_K - \boldsymbol{w}^{hp}\Big|_K \Big\|_{s,K} \leq C \frac{h_K^{\mu - s}}{p_K^{r-s}} \left\| \boldsymbol{v} \right\|_{r,K} \tag{3.19}$$

3. $\forall\ q^{hs} \in M^{hs}(\Omega)$, (3.16) *holds with p_K replaced by s_K*

4 FINITE ELEMENT FORMULATIONS

4.1 *hp*–Element Strategies

A semi–discrete *hp*-finite element approximation of the Navier-Stokes equations consists of seeking velocity and pressure approximations

$$\boldsymbol{u}^{hp}(t) \in V^{hp}(\Omega)\ ,\ P^{hs}(t) \in M^{hs}(\Omega)$$

for $t \in [0,T]$ such that

$$\boxed{\begin{array}{lcl} \left(\boldsymbol{u}_t^{hp}(t),\boldsymbol{\chi}\right) & + & c\left(\boldsymbol{u}^{hp}(t),\boldsymbol{u}^{hp}(t);\boldsymbol{\chi}\right) \\ +a\left(\boldsymbol{u}^{hp}(t),\boldsymbol{\chi}\right) & - & b\left(\boldsymbol{\chi},P^{hs}(t)\right) = L(\boldsymbol{\chi}) \\ & & b\left(\boldsymbol{u}^{hp}(t),\varphi\right) = 0 \\ & & \forall\, \boldsymbol{\chi} \in V^{hp}(\Omega), \varphi \in M^{hs}(\Omega) \end{array}} \tag{4.1}$$

where $V^{hp}(\Omega)$ and $M^{hs}(\Omega)$ are defined in (3.18).

To proceed further, the following issues must be resolved:

i) *Temporal Approximation*; the discretization of $[0,T]$ and the method of approximating the time–variation of $\boldsymbol{u}^{hp}$ and P^{hs}.

ii) *Convection Terms*; methods for handling the nonlinear convection effects embodied in the form $c(\cdot\ ,\ \cdot\ ;\ \cdot)$.

iii) *Pressure Approximations*; methods for constructing stable approximations on *hp*-meshes.

iv) *Diffusion Terms*; methods for handling the diffusion terms, $a(\cdot\ ,\ \cdot)$.

In the present exposition, these issues are dealt with as follows:

i) *High–Order, Semi–Implicit Splitting Schemes.* High–order temporal approximations are introduced to balance the high–order spatial approximations possible in adaptive *hp*-methods. Splitting methods are used so that convection steps are handled using explicit (or "weakly" implicit) methods, whilst diffusion steps are handled by a linear (parallelizable) implicit step. These are discussed in Section 5.

ii) *Exterior Penalty with Pressure Correction.* A non–singular, perturbed Lagrangian method is used to approximate element pressures which is essentially a regularized, discontinuous, exterior–penalty type approximation of the pressure P. In this way, the local

spectral orders s of the element pressures are determined by the quadrature rule chosen to integrate the bilinear form $b(\cdot\ ,\ \cdot)$. The regularization (penalty) parameter $\varepsilon = \varepsilon_K$ depends upon h_K and p_K and can vary element to element. The incompressibility condition, $div\ \boldsymbol{u} = 0$, is thus, relaxed, with $P^{hs}(\boldsymbol{\xi}_\ell) = -\varepsilon_K^{-1} div\ \boldsymbol{u}^{hp}(\boldsymbol{\xi}_\ell)$ at quadrature points $\boldsymbol{\xi}_\ell \in K$. Whenever $\|div\ \boldsymbol{u}^{hp}\|_{0,K}$ exceeds a preset tolerance, the pressure is globally corrected. These details are discussed more fully in the next subsection.

4.2 Pressure Approximations

We begin with a brief review of the concepts behind non-singular perturbations in the continuity equation as a device for handling pressure approximations.

It is well known that a notorious difficulty in the numerical solution of the Navier-Stokes equations is the treatment of the pressure approximation in a way that results in a stable scheme. At the heart of this issue is the LBB-condition, originally established for the Stokes problem by Ladyszhenskaya [48], which establishes that if the Stokes problem is to be well posed, the bilinear form $b: V \times M \rightarrow \mathbb{R}$ of (2.3) must be such that a constant $\beta > 0$ exists such that

$$\sup_{\boldsymbol{v} \in \mathring{\boldsymbol{V}} \backslash \boldsymbol{0}} \frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_1} > \beta \|q\|_0 \qquad \forall\, q \in M \tag{4.2}$$

A fairly large literature exists on ways to ensure the fulfillment of this condition for certain discrete problems (for summaries, see Carey and Oden [49], Kikuchi and Oden [50], and Brezzi and Fortin [51]).

Focusing on the Stokes problem (2.5) for simplicity, it is observed that the solution $(\boldsymbol{u}, p)$ is a saddle point of the Lagrangian

$$\left.\begin{aligned} L :\ & V \times M \rightarrow \mathbb{R} \\ & L(\boldsymbol{v}, q) = J(\boldsymbol{v}) - b(\boldsymbol{v}, q) \\ & J(\boldsymbol{v}) = \frac{1}{2} a(\boldsymbol{v}, \boldsymbol{v}) - L(\boldsymbol{v}) \end{aligned}\right\} \tag{4.3}$$

A unique saddle point of L exists whenever the energy functional $J(\boldsymbol{v})$ is coercive, convex, and differentiable (with respect to $\boldsymbol{v}$) for fixed q (which is always the case for the class of problems considered here), when $L(\boldsymbol{v}_0, q)$ is concave and differentiable with respect to q for fixed $\boldsymbol{v} = \boldsymbol{v}_0$ (which also holds), and when the following coercivity condition holds for some $\boldsymbol{v} = \boldsymbol{v}_0$ (see Oden [52, p. 81]):

$$\lim_{\|q\|_0 \rightarrow \infty} L(\boldsymbol{v}_0, q) \longrightarrow -\infty \tag{4.4}$$

To enforce this coercivity condition, and ensure a well-posed problem for evaluating the pressure, one can *perturb the Lagrangian* by the addition of terms that guarantee that (4.4) holds, e.g.,

$$L_\varepsilon(\boldsymbol{v}, q) = L(\boldsymbol{v}, q) + D_\varepsilon(\boldsymbol{v}, q) \tag{4.5}$$

where $D_\varepsilon(\boldsymbol{v}, q)$ is designed so that $L_\varepsilon(\cdot\ ,\ \cdot)$ satisfies (4.4). Then there exists a sequence of saddle points $(\boldsymbol{u}_\varepsilon, P_\varepsilon)$ such that

$$L_\varepsilon(\boldsymbol{u}_\varepsilon, q) \leq L_\varepsilon(\boldsymbol{u}_\varepsilon, P_\varepsilon) \leq L_\varepsilon(\boldsymbol{v}, P_\varepsilon) \qquad \forall\, (\boldsymbol{v}, q) \in V \times M \tag{4.6}$$

and one structures $D_\varepsilon(\cdot\ ,\ \cdot)$ so that $(\boldsymbol{u}_\varepsilon, P_\varepsilon) \to (\boldsymbol{u}, P)$ as $\varepsilon \to 0$, $(\boldsymbol{u}, P)$ being a solution of (2.5).

For the Stokes problem, one of the most common forms of the perturbation term in (4.5) is

$$D_\varepsilon(\boldsymbol{v}, q) = -\frac{\varepsilon}{2}\|q\|_0^2 \tag{4.7}$$

Then (4.4) holds for any $\varepsilon > 0$ and the perturbed form of the Stokes problem is

$$\left.\begin{array}{rcrcll} a(\boldsymbol{u}_\varepsilon, \boldsymbol{v}) & - & b(\boldsymbol{v}, P_\varepsilon) & = & L(\boldsymbol{v}) & \forall\, \boldsymbol{v} \in V \\ (\varepsilon P_\varepsilon, q) & + & b(\boldsymbol{u}_\varepsilon, q) & = & 0 & \forall\, q \in M \end{array}\right\} \tag{4.8}$$

which is formally equivalent to an exterior penalty approximation of P_ε, since $(4.8)_2$ yields

$$P_\varepsilon = -\varepsilon^{-1} div\ \boldsymbol{u}_\varepsilon \tag{4.9}$$

With (4.8), the momentum equation becomes

$$a(\boldsymbol{u}_\varepsilon, \boldsymbol{v}) + \varepsilon^{-1}(div\ \boldsymbol{u}_\varepsilon, div\ \boldsymbol{v}) = L(\boldsymbol{v}) \qquad \forall\, \boldsymbol{v} \in V \tag{4.10}$$

In discrete approximations of (4.9), we use numerical quadrature rules for evaluating the

integral of the penalty terms. For the master element $\widehat{K}$,

$$\int_{\widehat{K}} f g d\xi \approx \hat{I}(f,g) = \sum_{\ell=1}^{N_\ell} w_\ell f(\boldsymbol{\xi}_\ell)\, g(\boldsymbol{\xi}_\ell) \tag{4.11}$$

where w_ℓ are the quadrature weights and $\boldsymbol{\xi}_\ell$ are the quadrature points, and for affine–equivalent elements (recall (3.15)), we have

$$I(f,g) = \sum_{K \in \mathcal{T}_h} I_K(f,g); \qquad I_K(f,g) = \det \boldsymbol{T}_K \hat{I}(f,g) \tag{4.12}$$

The space of *discontinuous element pressures* $M^{hs}(\Omega)$ is defined by

$$M^{hs}(\Omega) = \left\{ q^{hs} \in M \left| q^{hs} \right|_K \circ F_K = \hat{q} \in Q^s\left(\widehat{K}\right),\ \hat{I}(\hat{q},\hat{q}) = \|\hat{q}\|^2_{L^2(\widehat{K})} \right\} \tag{4.13}$$

Thus, the pressures are discontinuous polynomials of degree s in each variable, with s determined by the quadrature order of $\hat{I}(\cdot\,,\,\cdot)$. For an hp-approximation $\boldsymbol{u}_\varepsilon^{hp}$ of velocities, the discrete–penalty formulation becomes,

$$a\left(\boldsymbol{u}_\varepsilon^{hp}, \boldsymbol{v}\right) + \varepsilon^{-1} I\left(div\ \boldsymbol{u}_\varepsilon^{hp}, div\ \boldsymbol{v}\right) = L(\boldsymbol{v}) \quad \forall\ \boldsymbol{v} \in V \tag{4.14}$$

and the pressure approximation is given by

$$P_\varepsilon^{hp} \in Q^h(\Omega);\ P_\varepsilon^{hp} \circ F_K(\boldsymbol{\xi}_\ell) = -\varepsilon^{-1} div\ \hat{\boldsymbol{u}}_\varepsilon^{hp}(\boldsymbol{\xi}_\ell) \tag{4.15}$$

The discrete LBB condition for this type of approximation is of the form,

$$\sup_{\boldsymbol{v}^{hp}\in V^{hp}\backslash\mathbf{0}} \frac{I\left(div\ \boldsymbol{v}^{hp}, q\right)}{\|\boldsymbol{v}^{hp}\|_{1,\Omega}} \geq \beta_{hp} \|q\|_{0,\Omega} \quad \forall\, q \in M^{hs}(\Omega) \tag{4.16}$$

A study of LBB conditions of this type for high–order *hp*-finite element methods has been conducted by Wu and Oden [53] for some special two–dimensional cases. The following result suggests how a stable approximation of pressure can be obtained.

Let Ω denote a rectangular domain in $\mathbb{R}^2$ that is partitioned into a mesh of quadrilateral elements K which are affine equivalent to a master element $\widehat{K} = [-1,1]^2$. Let the velocity approximation of Ω belong to the space,

$$V^{hp}(\Omega) = \left\{\boldsymbol{v}^{hp} \in \left(C^0(\Omega)\right)^2 \left|\boldsymbol{v}^{hp}\right|_K \in \left(Q^p(K)\right)^2\right\} \tag{4.17}$$

where $Q^p(K)$ is the space of hierarchical polynomial shape functions of degree p in x_1 and x_2, and let the pressure approximation q^{hs} belong to

$$M^{hs}(\Omega) = \left\{q^{hs} \in L^2(\Omega) \left|q^{hs}\right|_K \in Q^s(K)\right\} \tag{4.18}$$

Then the discrete LBB condition (4.15) is satisfied with β_{hp} independent of h and p whenever

$$p \geq 2,\ s = p - r,\ r \geq 2$$

∎

Numerical experiments suggest that this result also holds for three–dimensional cases.

4.3 Pressure Corrections

A great deal of attention has been given to pressure–based schemes for treating the incompressible Navier-Stokes equations in recent years. These families of methods originated in the work of Chorin [54], were the basis of the Simple and Simpler finite difference algorithms

of Patankar and Spalding [55] (see also [56]), and were studied in great detail within the context of finite elements by Gresho (e.g., [57,58]); see also Ramaswamy, et al. [59].

The ideas are based on extensions of the Helmholtz theorem:

> *For any vector field* $\boldsymbol{u}: \Omega \to \mathbb{R}^3$, *there exist scalar and vector potentials* φ *and* $\boldsymbol{\psi}$ *such that*
>
> $$\boldsymbol{u} = \boldsymbol{\nabla}\varphi + \text{curl } \boldsymbol{\psi}$$

Since curl $\boldsymbol{\nabla}\varphi = 0$ and div curl $\boldsymbol{\psi} = 0$, $\boldsymbol{u}$ can be decomposed into a vorticity-free field ($\boldsymbol{\nabla}\varphi \in H^{\perp}$) plus a divergence-free field (curl $\boldsymbol{\psi} \in H$). More generally, if Π denotes a projection of V into H (recall (2.2)), a projection $\Gamma : V \to H^{\perp}$ exists such that $\Gamma = I - \Pi$ (see Temam [45]).

Now suppose that during a numerical simulation, the approximate velocity field $\boldsymbol{u}^{hp}$ wanders outside the space H of divergence-free fields. We may then add to $\boldsymbol{u}^{hp}$ a "correction" $\boldsymbol{w}^{hp} \in H^{\perp}$ so that

$$\boldsymbol{u}^{hp} + \boldsymbol{w}^{hp} \in H \ , \ \textit{or} \ \ \textit{div } \boldsymbol{u}^{hp} + \textit{div } \boldsymbol{w}^{hp} = 0$$

But since $\boldsymbol{w}^{hp} \in H^{\perp}$, there exists a scalar potential φ such that $\boldsymbol{w}^{hp} = \boldsymbol{\nabla}\varphi$. Hence,

$$\Delta\varphi = -\textit{div } \boldsymbol{u}^{hp} \tag{4.19}$$

where $\Delta = \boldsymbol{\nabla}\cdot\boldsymbol{\nabla}$ is the Laplacian. The scalar φ is termed a *pressure correction*, since a check with the projection Γ of the discretized momentum equations into $H^{\perp}$ will show that $\boldsymbol{\nabla}\varphi$ is related to $\boldsymbol{\nabla}P$ according to relations of the type, $\boldsymbol{\nabla}\varphi - \Delta t\boldsymbol{\nabla}P = O(\Delta t^2)$, Δt being a time step. To (4.19) must be appended boundary conditions, $\partial\varphi/\partial n = \sigma$, where $\sigma = \sigma(h, \Delta t)$ is selected for consistency with (2.1) and the order of approximation used in its discretization in space and time (see [58]).

In the present work, we shall apply a pressure correction of the form (4.19) only when the departure of $\boldsymbol{u}^{hp}$ from H exceeds a preset tolerance e_{TOL}. Thus, if

$$\left\| \textit{div } \boldsymbol{u}^{hp} \right\|_{0,K} \geq e_{TOL} \ , \ K \in \mathcal{T}_h$$

we set

$$-\Delta\hat{P} = \Delta t^{-1} \textit{div } \boldsymbol{u}^{hp}$$

and, then,

$$P_{\text{corrected}} = P^{hs} + \hat{P}$$

5 HIGH–ORDER, SEMI–IMPLICIT SOLVERS

The regularized Navier-Stokes equations can be written in the form,

$$\frac{D\boldsymbol{u}}{Dt} = \boldsymbol{F}(\boldsymbol{u}) + \nabla \boldsymbol{P} \tag{5.1}$$

where

$$\begin{aligned} &\frac{D\boldsymbol{u}}{Dt} = \boldsymbol{u}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = \textit{the material time derivative of the velocity} \\ &\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{f} + 2\nu \nabla \cdot \boldsymbol{D}(\boldsymbol{u}) = \textit{diffusion vector} \end{aligned} \tag{5.2}$$

We shall outline several high–order splitting schemes. For a survey of splitting methods, see Marchuk [60].

5.1 High–Order Method of Characteristics

The classical method of characteristics (though not actually a splitting scheme) can be generalized to produce a remarkably effective solver. Recall that the motion of a material fluid particle $\boldsymbol{X}$ is given by the map

$$\boldsymbol{x} = \boldsymbol{\chi}(\boldsymbol{X}, t)$$

where $\boldsymbol{x}$ is the spatial position of $\boldsymbol{X}$ at time t and $\boldsymbol{\chi}$ is a smooth invertible function for every $t \in [0, T]$. The characteristics of (5.1) are determined by the trajectories,

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{u} \tag{5.3}$$

and $\boldsymbol{u}$ can be written as a function of particles $\boldsymbol{X}$ and time t in a material (Lagrangian) description of the motion:

$$\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X}, t), t) = \hat{\boldsymbol{u}}(\boldsymbol{X}, t); \quad \frac{D\boldsymbol{u}}{Dt} = \frac{\partial \hat{\boldsymbol{u}}}{\partial t}$$

Thus, over a time interval $(n\Delta t, (n+1)\Delta t)$,

$$\begin{aligned}
\hat{\boldsymbol{u}}^{n+1}(\boldsymbol{X}) &= \hat{\boldsymbol{u}}^{n}(\boldsymbol{X}) + \int_{t_n}^{t_{n+1}} [\boldsymbol{F}(\hat{\boldsymbol{u}}(\boldsymbol{X}, s)) + \nabla \hat{P}(\boldsymbol{X}, s)] ds \\
\boldsymbol{\chi}(\boldsymbol{X}, t_{n+1}) &= \boldsymbol{\chi}(\boldsymbol{X}, t_n) + \int_{t_n}^{t_{n+1}} \hat{\boldsymbol{u}}(\boldsymbol{X}, s) ds
\end{aligned}$$

where $\hat{\boldsymbol{u}}^{n+1}(\boldsymbol{X}) = \hat{\boldsymbol{u}}(\boldsymbol{X}, t_{n+1}) = \boldsymbol{u}(\boldsymbol{\chi}(\boldsymbol{X}, t_{n+1}), t_{n+1})$. The algorithm is thus divided into three steps:

1°. Determination of the Characteristics.

set

$$\begin{aligned}
\boldsymbol{X}^0 &= \boldsymbol{X}\ ,\ \boldsymbol{v}^0 = \hat{\boldsymbol{u}}(\boldsymbol{X}, t_{n+1}) = \boldsymbol{v}^0(\boldsymbol{X}^0, 0) \\
\Delta s &= (t_{n+1} - t_n)/N \\
do\ k &= 1,\ N \\
\boldsymbol{X}^k &= \boldsymbol{X}^{k-1} - \boldsymbol{v}^{k-1}\left(\boldsymbol{X}^{k-1}, (k-1)\Delta s\right)\Delta s \\
\boldsymbol{v}^k &= \hat{\boldsymbol{u}}\left(\boldsymbol{X}^k, t_{n+1} - k\Delta s\right) \\
&= \sum_{j=1}^{m} \boldsymbol{u}\left(\boldsymbol{x}^k, t_{n+1-j}\right) \varphi_j\left(t_{n+1} - k\Delta s\right) \\
end\ do
\end{aligned}$$

where the functions $\varphi_j = \varphi_j(t)$ are polynomial functions in t that interpolate $\boldsymbol{u}(\boldsymbol{x}^k, t)$ over past time steps.

To further reduce the error from the above procedure, we may introduce the following angle check in the loop:

Angle Check:

$$\text{If } \left|\cos^{-1}\left(\frac{\hat{\boldsymbol{v}}^k \cdot \hat{\boldsymbol{v}}^{k-1}}{\left\|\hat{\boldsymbol{v}}^k\right\| \left\|\hat{\boldsymbol{v}}^{k-1}\right\|}\right)\right| \leq \varepsilon_1 = \text{tol}_1, \text{ continue}$$

$$\text{else } \Delta s = \frac{1}{2}\Delta s$$

2^{o}. *Determination of* $\hat{\boldsymbol{u}}^{n+1}$

$$\boldsymbol{F}^k = \boldsymbol{F}^k\left(\boldsymbol{v}^k, t_{n+1} - k\Delta s\right)$$

$$\hat{\boldsymbol{u}}^{n+1}(\boldsymbol{X}) = \boldsymbol{u}\left(\boldsymbol{x}^N, t_n\right) + \sum_{k=0}^{N} w_k \boldsymbol{F}^k$$

$$\Longrightarrow \hat{\boldsymbol{u}}^{n+1}(\boldsymbol{X}) - w_0 \boldsymbol{F}\left(\hat{\boldsymbol{u}}^{n+1}(\boldsymbol{X})\right) = \boldsymbol{u}\left(\boldsymbol{x}^N, t_n\right) + \sum_{k=1}^{N} w_k \boldsymbol{F}^k$$

3^{o}. *Pressure–Continuity Check*

$$P_0^{n+1}\Big|_K = \sum_{\ell} P^{n+1}\left(\boldsymbol{\xi}_\ell\right) \varphi_\ell(\xi, \eta)$$

$$P^{n+1}\left(\boldsymbol{\xi}_\ell\right) = -\varepsilon^{-1} div\ \boldsymbol{u}^{n+1}\left(\boldsymbol{\xi}_\ell\right)$$

$$\beta_K \stackrel{\text{def}}{=} \|div\ \boldsymbol{u}^{n+1}\|_{0,K}^2$$

If $\beta_K \leq \varepsilon_2 = tol_2$, *go to* 1°,

else

$$P^{n+1}|_K = P^{n+1} + p^{n+1}$$

$$\int_K \boldsymbol{\nabla} p^{n+1} \boldsymbol{\nabla} \varphi\ dx = -\int_K div\ \boldsymbol{u}^{n+1} \varphi\ dx$$

$$\forall\ \varphi \in M^{hs}(K)$$

go to 3^{o}

end

Here the functions φ are the $Q^s(K)$ basis functions described earlier; other notations are self explanatory.

5.2 Multi–Step Methods

Multi–step methods, such as the Adams-Bashforth scheme, have been employed by several investigators concerned with spectral approximations of the Navier-Stokes equations (see, e.g., Canuto, Hussani, Quarteroni, and Zang [61], Koroczak and Patera [62], or Ramaswamy, Jue, and Akin [59]). A two–step scheme in this family of methods is indicated as follows:

1°. Explicit Step. An N-th order explicit Adams-Bashforth step is embodied in the expression,

$$\boldsymbol{u}^{n+1}(\boldsymbol{x}) = \boldsymbol{u}^{n}(\boldsymbol{x}) + \Delta t \sum_{k=0}^{N-1} \beta_k \boldsymbol{F}\left(\boldsymbol{u}^{n-k}\right) + O\left(\Delta t^{N+1}\right)$$

where the β_k are given in the following array:

Order	β_0	β_1	β_2	$\beta_3 \ldots$
1	1			
2	3/2	−1/2		
3	23/12	−16/12	5/12	
4	55/24	−59/24	37/24	−9/24

etc.

2°. Implicit Correction. The N-th order Adams-Moulton is expressed by

$$\boldsymbol{u}^{n+1}(\boldsymbol{x}) = \boldsymbol{u}^{n} + \Delta t \sum_{k=0}^{N-1} \overline{\beta}_k \boldsymbol{F}\left(\boldsymbol{u}^{n+1-k}\right) + O\left(\Delta t^{N+1}\right)$$

with

Order	$\overline{\beta}_0$	$\overline{\beta}_1$	$\overline{\beta}_2$	$\overline{\beta}_3$
2	1/2	1/2		
3	5/12	8/12	−1/12	
4	9/24	19/24	−5/24	1/24

The Adams-Bashforth/Adams-Moulton schemes can be used in a predictor–corrector algorithm with

$$\boldsymbol{u}^{*} = \boldsymbol{u}^{n} + \Delta t \sum_{k=0}^{N-1} \beta_k \boldsymbol{F}\left(\boldsymbol{u}^{n-k}\right) \qquad (predictor)$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^{n} + \Delta t \overline{\beta}_0 \boldsymbol{F}(\boldsymbol{u}^{*}) + \Delta t \sum_{k=1}^{N-1} \overline{\beta}_k \boldsymbol{F}\left(\boldsymbol{u}^{n+1-k}\right) \qquad (corrector)$$

5.3 Runge-Kutta Schemes

The classical explicit Runge-Kutta schemes provide another approach toward the development of high–order flow solvers. In general, these advance the solution in time according

to

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t \sum_{i=1}^{N} \alpha_i k_i$$

$$\sum_{i=1}^{N} \alpha_i = 1$$

$$k_i = \boldsymbol{F}\left(\boldsymbol{u}^n + \Delta t \sum_{j=1}^{m} a_{ij} k_j\right)$$

with $1 \leq m \leq i-1$ for explicit schemes, $m = i$ for semi–implicit schemes, and $i+1 \leq m \leq N$ for fully implicit schemes. See Butcher [63] for full details and tables of the coefficients a_{ij}.

We compare results obtained with some of these schemes in Section 7.

6 ERROR ESTIMATION AND ADAPTIVITY

The *hp*-adaptive strategy must involve a running audit of the error in the approximate velocity and pressure fields. Existing mathematical theories for *a posteriori* error estimation primarily apply to simple elliptic, scalar boundary–value problems and few results exist for error estimation of solutions to the Navier-Stokes equations. A study of error estimation techniques for *hp*-finite element approximations of elliptic problems is given in [42]; some extensions of these ideas to time–dependent problems are discussed in [63] and error residual methods are adapted to calculations of compressible flow problems in [64].

In this investigation, we extend the error residual methods of Ainsworth and Oden [65] to incompressible Navier-Stokes equations by exploiting the particular structure of the approximations inherent in the splitting algorithms described in the preceding section.

6.1 A Posteriori Error Estimators

The $(n+1)st$-step of a semi–explicit method of the type discussed in Section 5 requires the solution of a linear elliptic system of the form,

$$\begin{aligned} \boldsymbol{u}^{n+1} \; - \; & 2\Delta t \nabla \cdot \nu \boldsymbol{D}\left(\boldsymbol{u}^{n+1}\right) - \Delta t \varepsilon^{-1} \nabla \left(\nabla \cdot \boldsymbol{u}^{n+1}\right) \\ = \; & \Delta t \boldsymbol{f}^{n+1} + \boldsymbol{u}^{n+\frac{1}{2}} \end{aligned} \tag{6.1}$$

with, for example,

$$\boldsymbol{u}^{n+\frac{1}{2}} = \boldsymbol{u}^n\left(\boldsymbol{x}^N\right)$$

for the method of characteristics. Setting

$$\boldsymbol{u}^{n+1} = \boldsymbol{u} \; ; \; \boldsymbol{AD}(\boldsymbol{u}) = \Delta t \left(2\nu \boldsymbol{D}(\boldsymbol{u}) + \varepsilon^{-1} \mathbf{1} \nabla \cdot \boldsymbol{u}\right)$$

$$\tilde{\boldsymbol{f}} = \Delta t \boldsymbol{f}^{n+1} + \boldsymbol{u}^{n+\frac{1}{2}},$$

the problem reduces to the symmetric, linear, elliptic boundary–value problem,

$$\boxed{\begin{aligned} -\nabla \cdot \boldsymbol{A}\boldsymbol{D}(\boldsymbol{u}) + \boldsymbol{u} &= \tilde{\boldsymbol{f}} \quad \text{in} \quad \Omega \\ \boldsymbol{u} &= \boldsymbol{0} \quad \text{on} \quad \Gamma_1 \\ \boldsymbol{n} \cdot \boldsymbol{A}\boldsymbol{u} &= \boldsymbol{g} \quad \text{on} \quad \Gamma_2 \end{aligned}} \tag{6.2}$$

Here $\boldsymbol{A}$ is the fourth order tensor, $A_{ijk\ell} = \Delta t(2\nu\delta_{i\ell}\delta_{jk} + 2\nu\delta_{jk}\delta_{j\ell} + \varepsilon^{-1}\delta_{ij}\delta_{k\ell})$. The corresponding bilinear and linear forms are

$$\begin{aligned} &\tilde{a} : V \times V \to \mathbb{R} \,, \; \tilde{L} : V \to \mathbb{R} \\ &\tilde{a}(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \left(\boldsymbol{D}(\boldsymbol{v})^T \boldsymbol{A}\boldsymbol{D}(\boldsymbol{u}) + \boldsymbol{u} \cdot \boldsymbol{v} \right) dx = \sum_{K \in \mathcal{T}_h} \tilde{a}_K(\boldsymbol{u}, \boldsymbol{v}) \end{aligned} \tag{6.3}$$

$$\tilde{L}(\boldsymbol{v}) = \int_\Omega \tilde{\boldsymbol{f}} \cdot \boldsymbol{v} \; dx + \int_{\Gamma_2} \boldsymbol{g} \cdot \boldsymbol{v} \; ds = \sum_{K \in \mathcal{T}_h} \tilde{L}_K(\boldsymbol{v}) \tag{6.4}$$

Here $\tilde{a}_K(\cdot \,,\, \cdot)$ and $\tilde{L}_K(\cdot)$ are the bilinear and linear forms defined on a partition $\mathcal{T}_h$ of Ω (recall (2.6), (2.7)). We define the local and global energy norms by

$$\|\boldsymbol{u}\|_E = \sqrt{\tilde{a}(\boldsymbol{u}, \boldsymbol{u})} \,, \; \|\boldsymbol{u}\|_{E,K} = \sqrt{\tilde{a}_K(\boldsymbol{u}, \boldsymbol{u})} \tag{6.5}$$

Thus

$$\|\boldsymbol{u}\|_E^2 = \sum_{K \in \mathcal{T}_h} \|\boldsymbol{u}\|_{E,K}^2$$

We wish to estimate the approximation error of $\boldsymbol{e}(\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t) - \boldsymbol{u}^{hp}(\boldsymbol{x}, t)$ at $t = (n+1)\Delta t$ to within term of $O(\Delta t^2)$, in the energy norms (6.5).

We next introduce the following notations and conventions (Cf [65]):

- *The partition $\mathcal{T}_h$ of Ω has $N = N(h)$ elements $K \equiv \Omega_K$, $\overline{\Omega} = \bigcup\{K \in \mathcal{T}_h\} = \bigcup_{K=1}^{N} \overline{\Omega}_K$ (thus "K" is used to indicate both a subdomain and an index of the subdomain)*

- $\Gamma_{KL} = \partial\Omega_K \bigcap \partial\Omega_L$, $0 \le K, L \le N$ *are sets consisting of a finite number* $\rho(K,L)$ *of segments such that*

$$\Gamma_{KL} = \bigcup_{M=1}^{\rho(K,L)} \Gamma_{KL}^M \bigcup \Sigma_{KL}$$

$$\Sigma_{KL} = \textit{isolated (nodal) points } \left(\Sigma_{KL} \bigcap \Gamma_{KL} = \emptyset\right)$$

- $\Gamma_{0K} = \partial\Omega_K \bigcap \partial\Omega$, $\partial\Omega_K = \bigcup_{L,M} \overline{\Gamma}_{KL}^M$

- $E = E(\mathcal{T}_h)$ =*boundary segments of the partition* $\mathcal{T}_h$

$$= \bigcup_{\substack{K,L=0 \\ K>L \\ 1\le M\le \rho(K,L)}}^{N} \Gamma_{KL}^M$$

- E_I = *segments lying interior to* $\Omega = \bigcup_{\substack{K,L=1 \\ K>L \\ 1\le M\le \rho(K,L)}}^{N} \Gamma_{KL}^M$

- $\sigma_{KL} = -\sigma_{LK} = \begin{cases} +1 & \text{if } K > L \\ -1 & \text{if } K < L \end{cases}$

- $\boldsymbol{n}(\boldsymbol{s}) = \sigma_{KL}\boldsymbol{n}_K(\boldsymbol{s}) = \sigma_{LK}\boldsymbol{n}_L(\boldsymbol{s})$, $\boldsymbol{s} \in \Gamma_{KL}^M$,
  thus $\boldsymbol{n}$ *is a unit normal pointing outward from the subdomain with largest index*

- $\alpha_{KL}^M : \Gamma_{KL}^M \to \mathbb{R}$; $\alpha_{KL}^M(\boldsymbol{s}) + \alpha_{LK}^M(\boldsymbol{s}) = 1$,
  $0 \le K, L \le N$, $1 \le M \le \rho(K,L)$

- *Jumps and convex combinations:*

$$[\![\boldsymbol{v}\cdot\boldsymbol{n}]\!]_{\Gamma_{KL}^M} = \sigma_{KL}\gamma\boldsymbol{v}_K \cdot \boldsymbol{n}_K - \sigma_{LK}\gamma\boldsymbol{v}_L \cdot \boldsymbol{n}_L$$

$$\langle \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\alpha|\Gamma_{KL}^M} = \alpha_{KL}^M\gamma\boldsymbol{v}_K \cdot \boldsymbol{n}_K - \alpha_{LK}^M\gamma\boldsymbol{v}_L \cdot \boldsymbol{n}_L$$

($\gamma\boldsymbol{v}_K$ = *trace of* $\boldsymbol{v} \in V(K)$ *onto* $\partial\Omega_K$, $[\![\boldsymbol{v}\cdot\boldsymbol{n}]\!]$ *is the jump in* $\boldsymbol{v}\cdot\boldsymbol{n}$ *across neighboring subdomains, while* $\langle \boldsymbol{v}\cdot\boldsymbol{n}\rangle_\alpha$ *is a linear weighted average of* $\boldsymbol{v}\cdot\boldsymbol{n}$ *between neighboring subdomains).*

The *element residual functions* are then defined by,

$$\begin{aligned} \boldsymbol{r}_K(\boldsymbol{x}) &= \tilde{\boldsymbol{f}}(\boldsymbol{x}) + \left(\boldsymbol{\nabla}\cdot\boldsymbol{AD}\left(\boldsymbol{u}^{hp}\right) - \boldsymbol{u}^{hp}\right)(\boldsymbol{x}) \ , \ \boldsymbol{x}\in K \\ \boldsymbol{R}_K(\boldsymbol{s}) &= \begin{cases} \boldsymbol{g} - \boldsymbol{n}_K\cdot\boldsymbol{AD}\left(\boldsymbol{u}^{hp}\right) & \boldsymbol{s}\in\Gamma_{K0}^M\bigcap\Gamma_2 \ , \ 1\le M\le\rho(K,0) \\ -\alpha_{KL}\left[\!\left[\boldsymbol{n}\cdot\boldsymbol{AD}\left(\boldsymbol{u}^{hp}\right)\right]\!\right] \ , & \boldsymbol{s}\in\Gamma_{KL}^M \ , \ 1\le K,L\le N \\ & 1\le M\le\rho(K,L) \end{cases} \end{aligned}$$

$$\boldsymbol{\delta}_K = \begin{cases} meas(K)^{-1}\left\{\int_K \boldsymbol{r}_K dx + \oint_{\partial K}\boldsymbol{R}_K ds\right\} \text{ if } \partial K\bigcap\partial\Omega = \emptyset \\ \\ \mathbf{0} \text{ if otherwise} \end{cases} \tag{6.6}$$

Each finite element is now endowed with a *complimentary energy* given by the functional

$$G_K(\boldsymbol{\sigma}) = -\frac{1}{2}\int_K \boldsymbol{\sigma}^T\boldsymbol{A}^{-1}\boldsymbol{\sigma}\ dx - \frac{1}{2}\int_K |\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \boldsymbol{r}_K|^2\, dx \tag{6.7}$$

Theorem 6.1 (Cf [60]). *Let* $\boldsymbol{e}$ *denote the approximation error*

$$\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{u}^{hp} \tag{6.8}$$

where $\boldsymbol{u}$ *is the exact solution to* (6.1) *and* $\boldsymbol{u}^{hp}$ *is its hp-finite element approximation. Then*

$$\|\boldsymbol{e}\|_E^2 \le -2\sum_{K\in\mathcal{T}_h} G_K(\boldsymbol{\sigma}) \tag{6.9}$$

for all $\boldsymbol{\sigma}\in W_K$, *where*

$$\begin{aligned} W_K &= \Big\{\sigma_{ij}(\boldsymbol{x})\in(L^2(K))^{n^2} \,|\, \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}\in(L^2(K))^n \ , \\ &\quad \sigma_{ij} = \sigma_{ji} \ , \ 1\le i,j\le 3 \ , \ \sigma_{ij}n_{Kj} = R_{Ki}(s), \\ &\quad \text{on } \partial K\setminus\Gamma_1\Big\} \end{aligned} \tag{6.10}$$

Defining,

$$\left.\begin{aligned} \eta_K^2(\boldsymbol{\sigma}) &= \varepsilon_K^2(\boldsymbol{\sigma}) + \Lambda_K^2(\boldsymbol{\sigma}) \\ \varepsilon_K^2(\boldsymbol{\sigma}) &= \int_K \boldsymbol{\sigma}^T \boldsymbol{A}^{-1} \boldsymbol{\sigma}\, dx \\ \Lambda_K^2(\boldsymbol{\sigma}) &= \int_K |\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \boldsymbol{r}_K|^2\, dx \end{aligned}\right\} \tag{6.11}$$

we see that

$$\|e\|_E^2 \leq \sum_{K \in \mathcal{T}_h} \eta_K^2(\boldsymbol{\sigma}) \qquad \forall\, \boldsymbol{\sigma} \in \prod_{K \in \mathcal{T}_h} W_K \tag{6.12}$$

Thus, the functionals $\eta_K^2(\boldsymbol{\sigma})$ provide a bound on the global error for any "pseudo stress" $\boldsymbol{\sigma}$. It remains only to construct an efficient way to calculate (or approximate) a suitable local field $\boldsymbol{\sigma}$.

One technique that leads to asymptotically exact estimates is as follows:

Let $\boldsymbol{U} \in V^{h,p+1}(K)$ be such that

$$\begin{aligned} \int_K \boldsymbol{D}(\boldsymbol{\chi})^T \boldsymbol{A} \boldsymbol{D}(\boldsymbol{U}) dx &= \int_K (\boldsymbol{r}_K + \boldsymbol{\delta}_K) \cdot \boldsymbol{\chi} dx \\ &+ \int_{\partial K \setminus \Gamma_1} \boldsymbol{R}_K \cdot \boldsymbol{\chi} ds + \int_{\partial K \cap \Gamma_2} \boldsymbol{g} \cdot \boldsymbol{\chi} ds \end{aligned} \tag{6.13}$$

$$\forall\, \boldsymbol{\chi} \in V^{h,p+1}(K)$$

It can be shown that there exist a unique $\boldsymbol{U}$ satisfying this equation. Set

$$\boldsymbol{\sigma}^* = \boldsymbol{A} \boldsymbol{D}(\boldsymbol{U}) \tag{6.14}$$

Then it can be shown that $\Lambda_K^2(\boldsymbol{\sigma}^*) \leq \Lambda_K^2(\boldsymbol{\sigma})$, $\forall\, \boldsymbol{\sigma} \in W_K$, and that $\eta_K^2(\boldsymbol{\sigma}^*)$ is, therefore,, an acceptable error indicator. We choose

$$\eta_K^2 = \int_K \boldsymbol{\sigma}^{*T} \boldsymbol{A}^{-1} \boldsymbol{\sigma}^* dx + meas(K) \delta_K^2 \tag{6.15}$$

6.2 Adaptive Strategy

The adaptive algorithm employed is briefly outlined is as follows:

1. Begin with an initial mesh, generally obtained after solving the Stokes problem with one or two h-refinements and uniform p, or with p higher in regions where effects of singularities are expected to be significant.
2. Solve the problem on the initial mesh.
3. Compute $\Delta\eta_K^s$ due to the increase of every unconstrained degree of freedom of element K.
4. Choose the change in p_K or h_K that produced the maximum change in error

$$\Delta\eta_K^* = \max_s \Delta\eta_K^s$$

   and redistribute h_K and p_K accordingly.
5. Go to item 2.

Details of this procedure are given in [43].

7 NUMERICAL EXPERIMENTS

In this section, we present selected results of several numerical experiments designed to test the methods and the general approach described earlier. These primarily concern standard model problems, such as the backstep channel problem and the driven cavity problem.

7.1 The Backstep Channel Problem in Two Dimensions

A standard benchmark problem is the so–called backstep channel problem depicted in Fig. 5. The inflow velocity is parabolic,

$$u_2 = 0, \qquad u_1 = \frac{4U_0}{a^2}\left(\frac{a^2}{4} - y^2\right)$$

and no–flow conditions are imposed at all other boundaries except the outflow boundary $x_0 = b + c$, where we set $\boldsymbol{\sigma}(\boldsymbol{u}(t),\ P(t)) \cdot \boldsymbol{n} = \mathbf{0}$. The recirculation lengths L_1, L_2 shown depend upon the Reynolds number, and data on the variation of L_1 and L_2 with Re has been obtained by several investigators using other schemes (see, in particular, Ghia, et al. [64]).

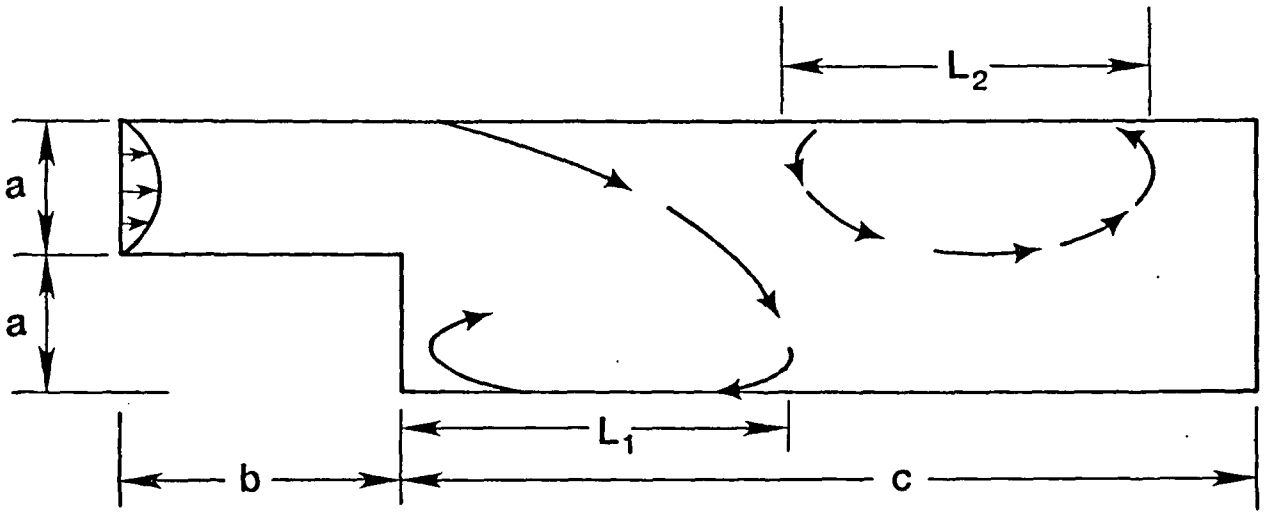


Figure 5: Geometry and notation for the two–dimensional backstep channel problem.

The first results cited here focus on our comparisons of the schemes described in Section 5. No attempt was made to optimize or vectorize the coding in these calculations, and a GMRES scheme was employed to solve the diffusion steps in all cases. The uniform mesh of Q^3/Q^1 elements shown in Fig. 6, corresponding to around 1200 degrees of freedom, was used.

Computed velocity profiles over a portion of the mesh are shown in Figs. 7 and 8. Figure 7 contains results obtained using a fourth–order Runge-Kutta (RK4) scheme. There we see computed velocity fields at $t = 2.00$ secs. obtained using a time step $\Delta t = 0.1$ sec. Figure 8 contains similar results for the same time step Δt but computed using a first–order MOC (method of characteristics) technique. The computed velocity fields are essentially the same, but were obtained by the MOC required only 40 percent of the CPU time of the RK4 technique.

Similar results for a $Re = 600$ are shown in Figs. 9–11. In Fig. 9, we see computed velocities delivered by the RK4 scheme at $t = 1.00$ secs. corresponding to $\Delta t = 0.1$ sec. A high–order (fourth–order) MOC scheme was also used to solve this problem and results are shown in Fig. 10. This calculation required approximately 75 percent of the CPU time used by the RK4 scheme. A fourth–order Adams-Bashforth scheme was also used to solve this example, but it failed for $\Delta t = 0.1$ sec. and was stable for $\Delta t \leq 0.05$ sec. Results obtained for this scheme at $t = 0.50$ sec. are shown in Fig. 11.

These calculations were repeated for $Re = 100$ and 1000, and a comparison of results is given in Table 1 below. Variances in time steps shown in the table are due to the conditional stability of the explicit steps in the RK4 scheme and the fourth order Adams-Bashorth scheme, which proved to be unstable at $\Delta t = 0.05$ sec. for $Re = 1000$. The method of characteristics appears to be unconditionally stable and to be superior to the other high–order schemes for these calculations. The relative efficiency of these schemes could, of course, change with appropriate optimization of the coding. All calculations were run on a single processor of the Alliant FX8 computer and are normalized with respect to the MOC for $Re = 100$. Results obtained using multiple processors are underway and will be reported in future work.

Table 1: Comparison of High–Order Schemes

Q^3/Q^1 Elements 1200 degrees of freedom

Re	300		600	
	Δt	CPU	Δt	CPU
RK4	0.1	2.43	0.1	2.44
Adams/B4	0.1	3.02	0.05	—
MOC4	0.1	1.00	0.1	1.04

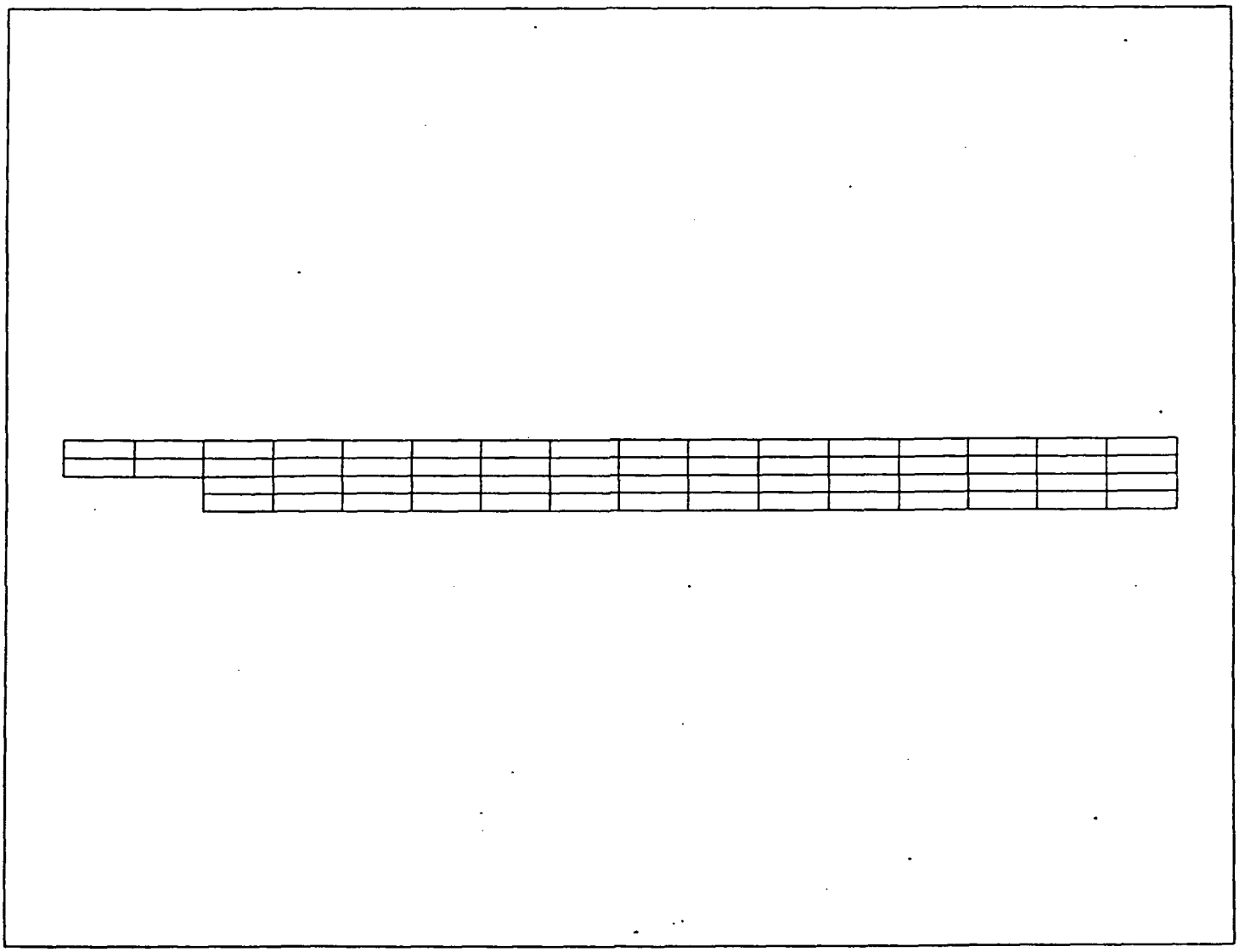


Figure 6: A uniform mesh of Q^3/Q^1 elements with 1202 degrees of freedom: $a/b = 2/7$, $c/a = 1/24$, $Re = 300$ and $Re = 600$.

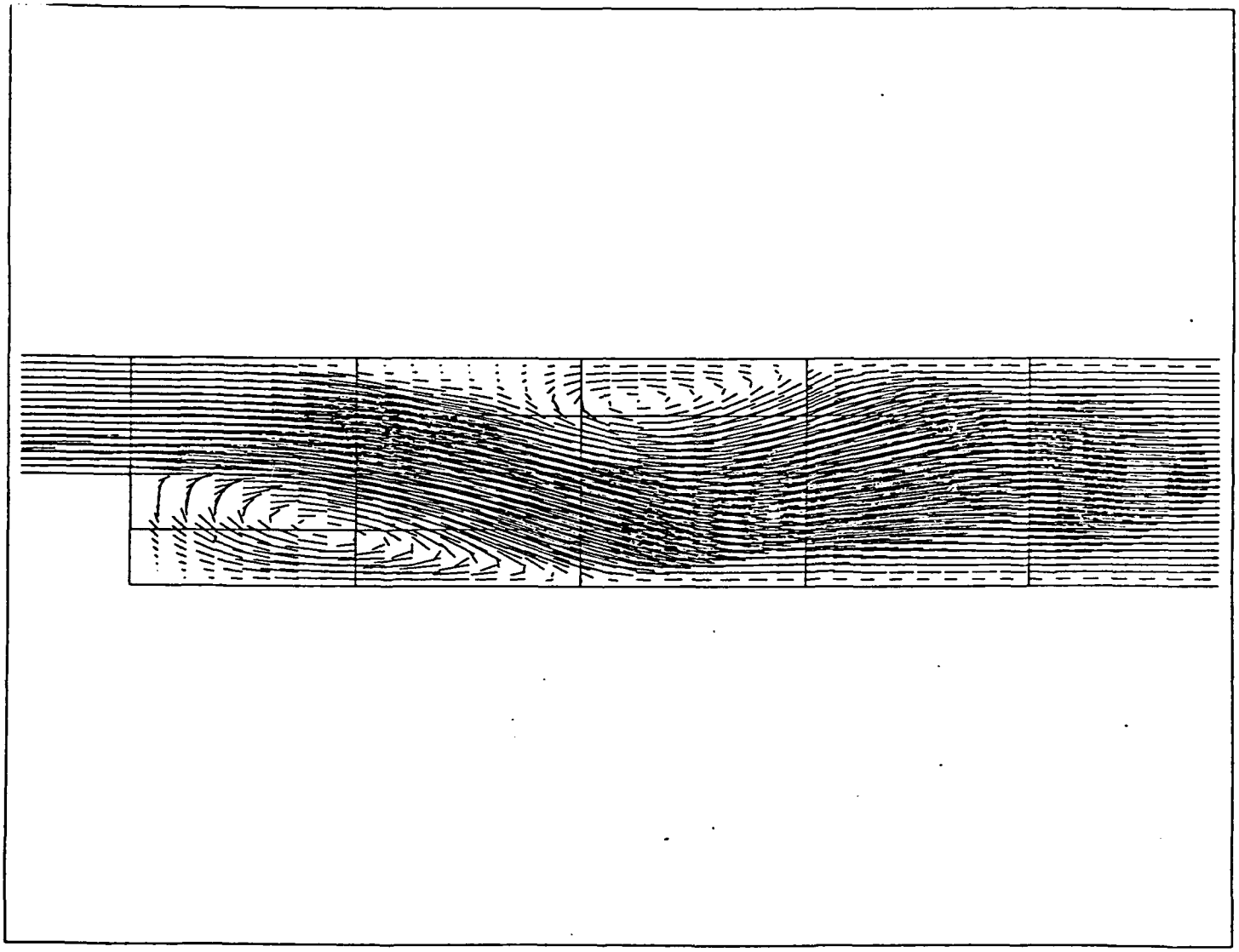


Figure 7: Velocity profiles: fourth-order Runge-Kutta scheme, Q^3/Q^1 elements, $\Delta t = 0.1$, $t = 2.0$ sec, $Re = 300$.

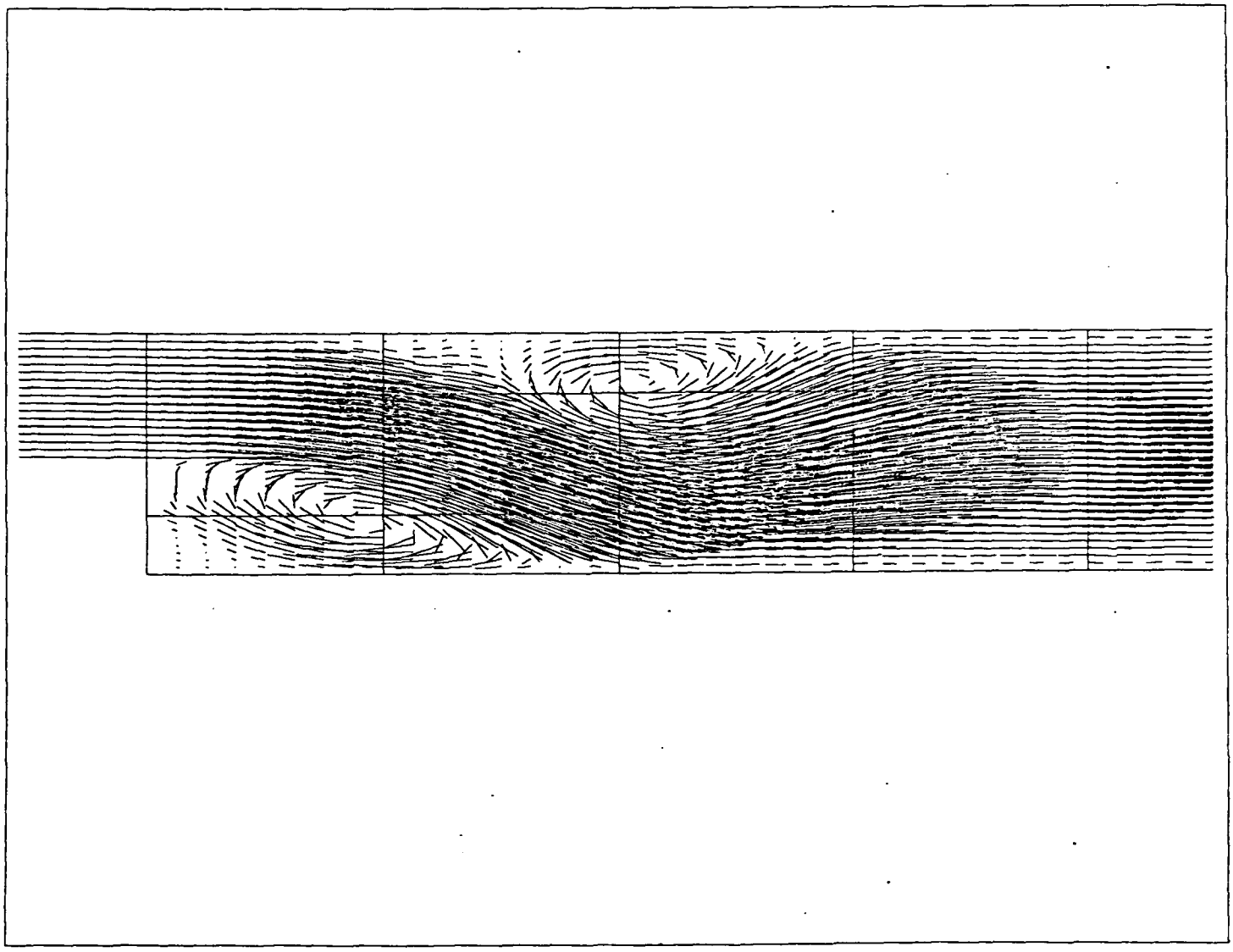


Figure 8: Velocity profiles: methods of characteristics, Q^3/Q^1, $\Delta t = 0.1$, $t = 2.00$ sec, $Re = 300$.

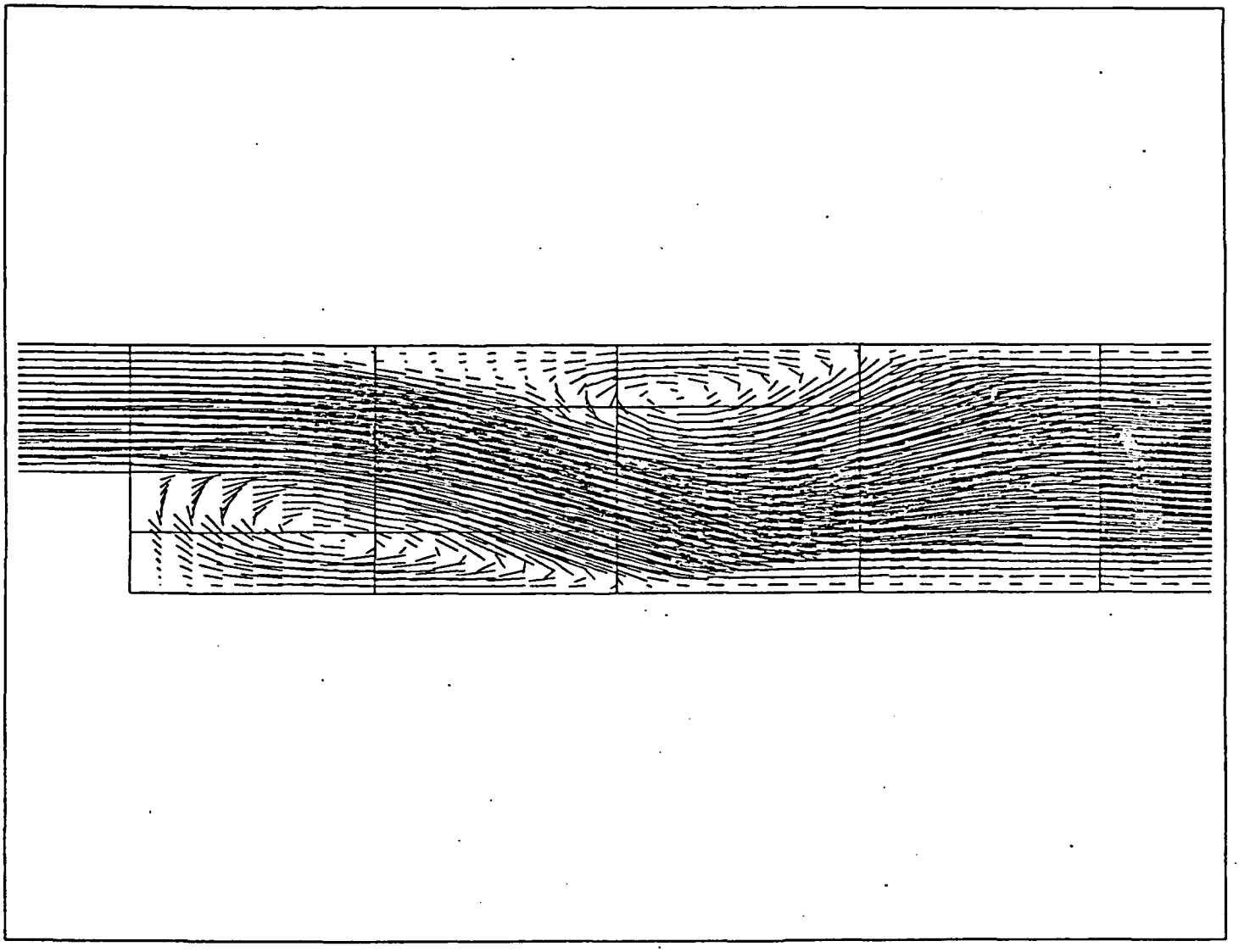


Figure 9: Computed velocity fields at $t = 1.00$ sec: $RK4$, $\Delta t = 0.1$ sec, $Re = 600$.

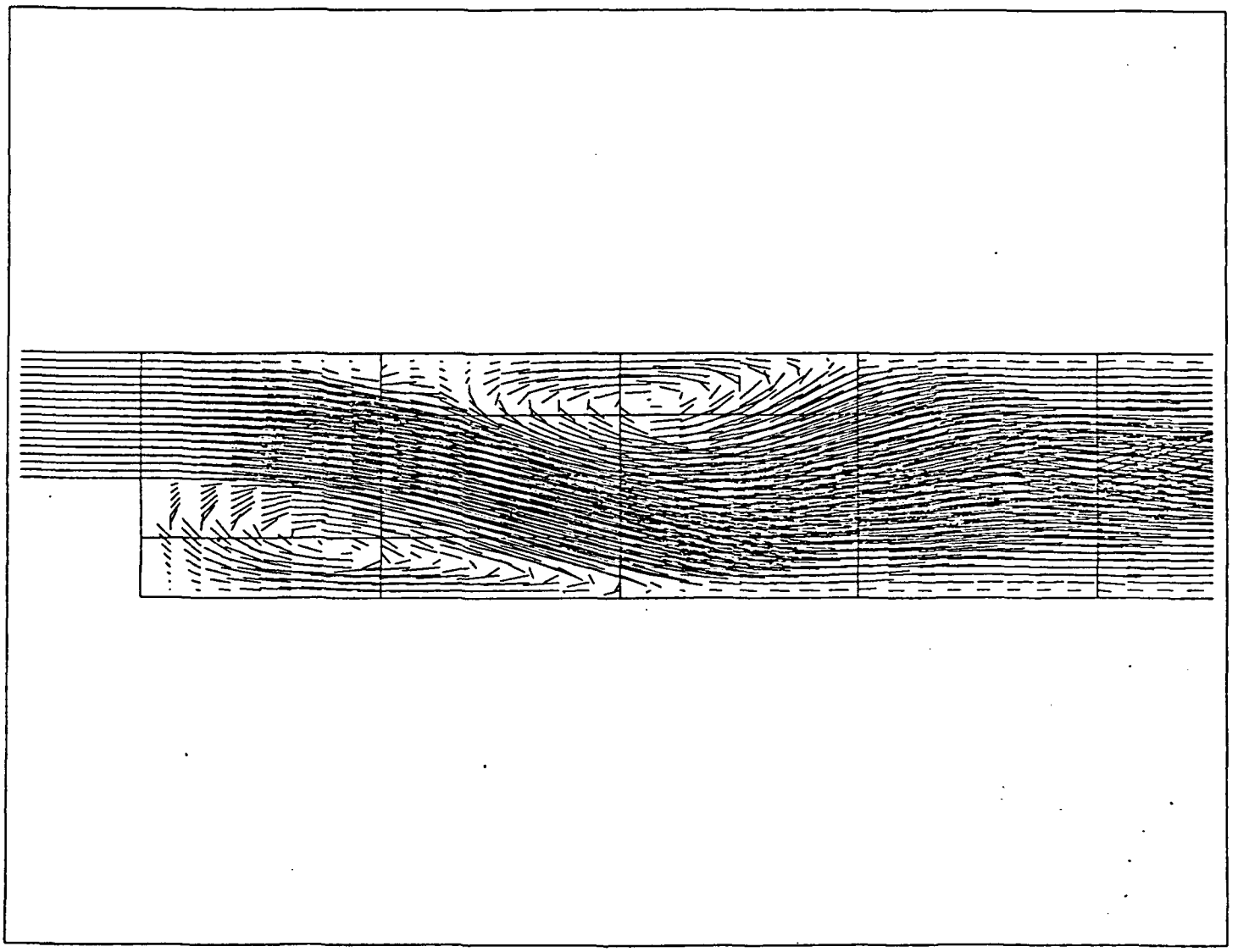


Figure 10: Computed velocity fields at $t = 1.00$ sec, fourth-order MOC, $\Delta t = 0.1$ sec, $Re = 600$.

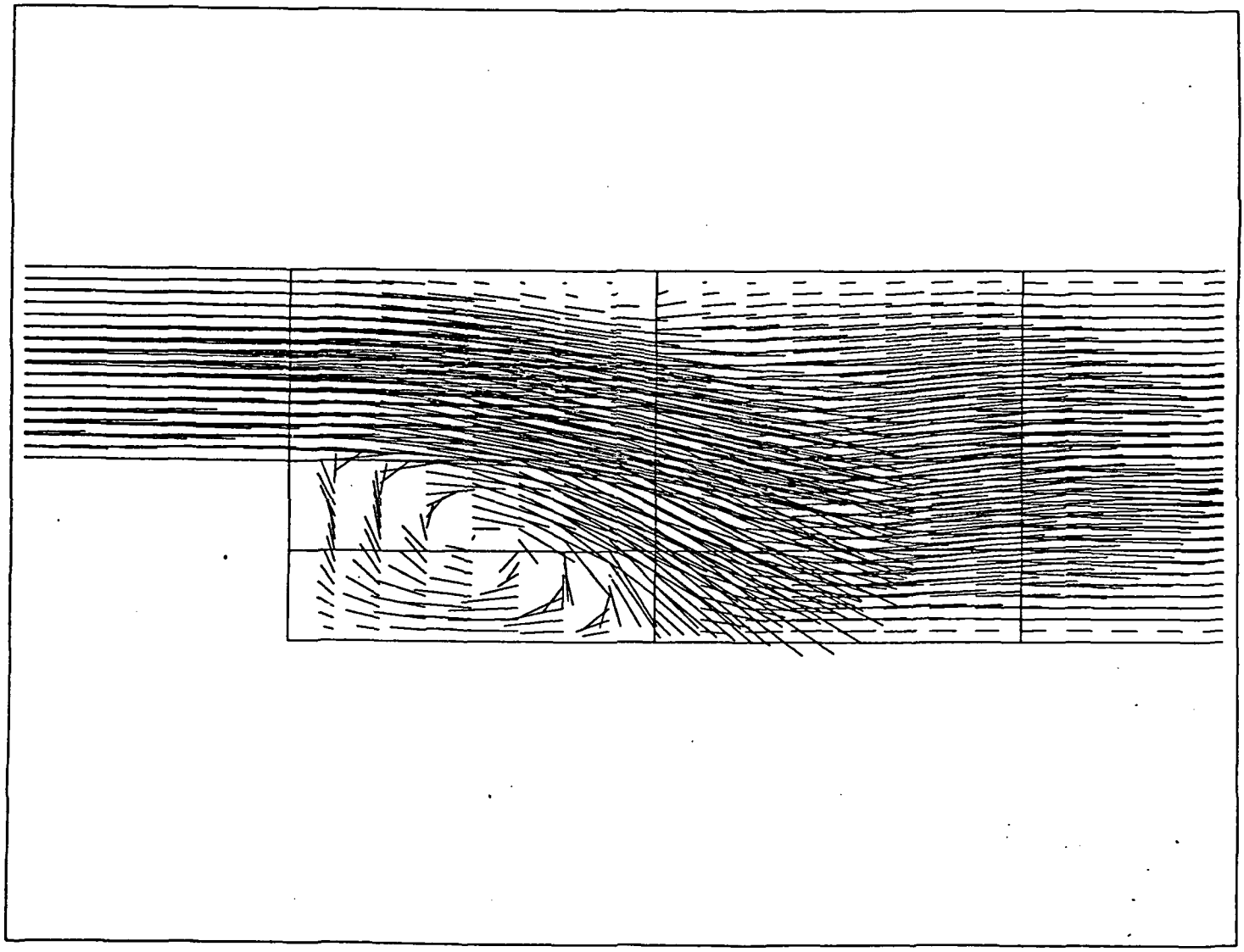


Figure 11: Computed velocity field at $t = 0.50$ sec: conditionally stable, fourth-order Adams-Bashforth scheme with $\Delta t = 0.05$ sec, $Re = 600$.

Q^4/Q^2 **Elements 1756 degrees of freedom**

Re	100		1000	
	Δt	CPU	Δt	CPU
RK4	0.05	6.60	0.01	6.15
Adams/B4	0.05	5.66	0.01	—
MOC4	0.1	1.00	0.1	2.57

Table 2 contains computed values of the reattachment length L_1 and L_2 of Fig. 5 computed using a fixed grid of Q^3/Q^1 elements for various Reynolds numbers for $a = 0.51485$ compared with those obtained by Ghia, et al. [64] using a 12,870 degrees of freedom finite difference calculation. These results correspond to solutions at $t = 100.0$ sec. and were obtained using only 1394 degrees of freedom.

Table 2: Reattachment Length Comparisons

Re	L_1		L_2	
	Present	Ref. [64]	Present	Ref. [64]
150	3.23	3.38	—	—
300	4.96	4.95	4.05	3.75
507	6.08	6.25	4.70	4.75
600	6.39	6.50	4.97	5.00

7.2 A Backstep Channel in Three Dimensions

Some preliminary results on a three–dimensional backstep channel calculation are displayed in Fig. 12 which contains u_1-velocity profiles for the Stokes problem, the solution of which is used as the initial conditions for integrating the Navier-Stokes equations. Here a uniform mesh of Q^2/Q^1 elements is used (it being understood that Q^r corresponds to tensor products of polynomials of degree r on a hexahedral element). Solutions for $Re = 300$, 40 elements, $\Delta t = 0.1$ sec. at $t = 5.0$ sec. are shown. A more detailed analysis of this problem is underway and is to be discussed in a future report.

7.3 Flow Around a Square Body, $Re = 250$

Figure 13 shows an initial hp mesh of cubic and quintic elements surround a square obstacle in a rectangular flow channel. A parabolic inflow velocity is prescribed, the intensity of which corresponds to a Reynolds number of 250 relative to the side length of the square. The finite

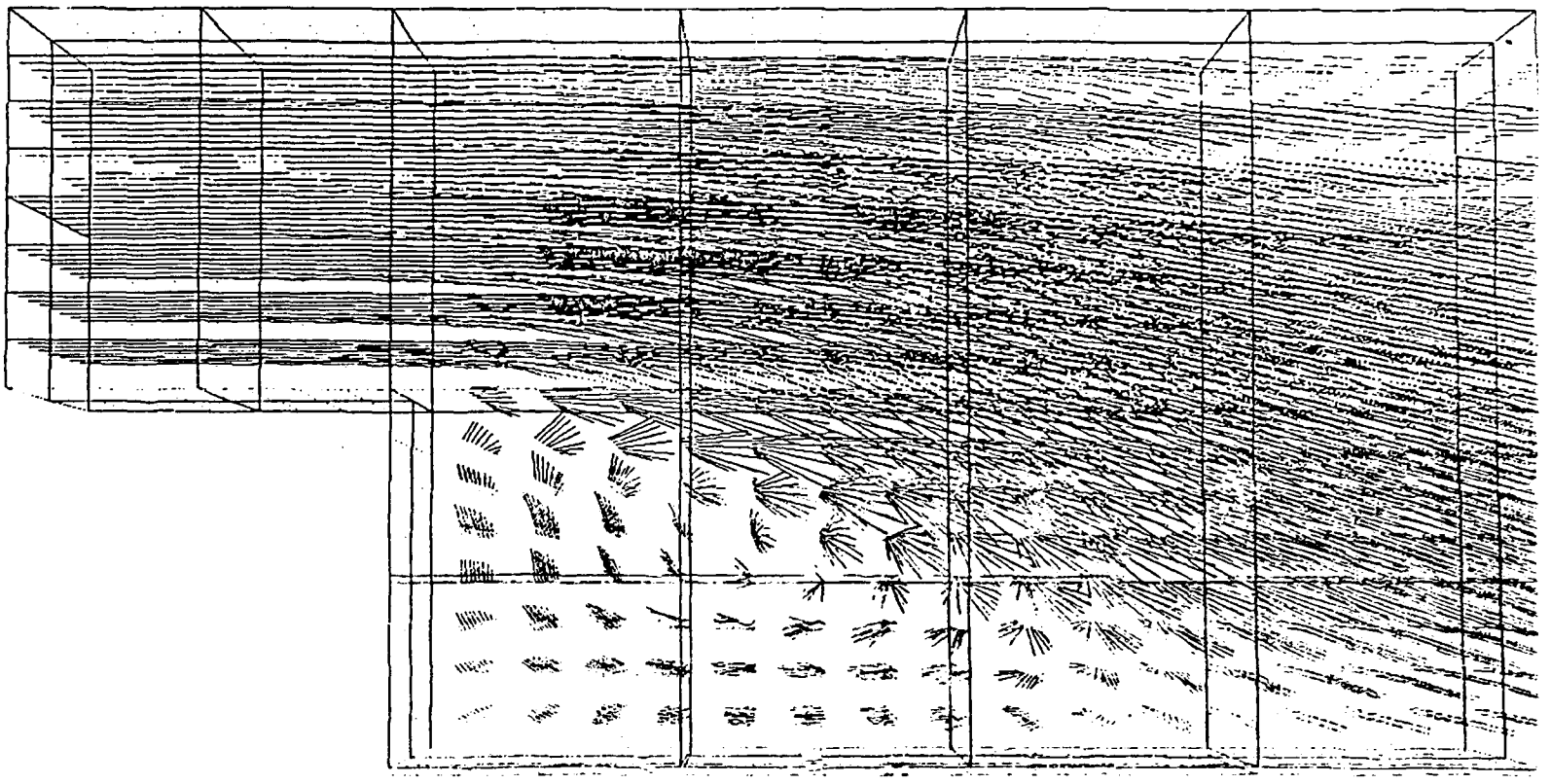


Figure 12: Computed velocity field in a three-dimensional backstep channel: $Re = 300$, $t = 5.0$ sec., MOC1.

element model contains only 1656 degrees of freedom. A second-order semi-implicit Runge-Kutta scheme was used to integrate the equations of motion, with penalty parameter ε set uniformly to 10^{-4}.

Computed velocity vectors and contours of the longitudinal component of velocity are shown in Fig. 14 at $t = 230$ sec. Contours of the u_2-velocity at this time are shown in Fig. 15.

7.4 Two–Dimensional Driven Cavity, $Re = 5000$

The classical driven cavity problem is next considered as a standard benchmark test. Two cases are presented. In the first case, only h-refinement is used with $p = 4$ uniformly over the square grid. The mesh is shown in Fig. 16, has 2393 degrees of freedom. The u_1-velocity of the "lid" is prescribed to be uniform and is set to correspond to a Reynolds number of 5000. The adaptive algorithm automatically refines the mesh near the corners to resolve the corner singularities.

In this test, the method of characteristics is used to integrate the solution to an apparent steady-state solution. Computed $u_1 = u$ and $v_1 = v$ velocity contours are shown in Fig. 17. Computed pressure contours are shown in Fig. 18 compared with those of Gresho and Chan [58], which were obtained using a fine graded mesh of bilinear elements. It is observed that excellent agreement is achieved. Notice that the plotting routine has detected the slight discontinuities in the pressure field, indicated by small but stable and non-oscillatory jumps at boundaries of the larger elements.

A second case was run using the nonuniform hp mesh shown in Fig. 19 which involved only 1442 degrees of freedom. In this case, the adaptive procedure produced the mesh shown with Q^5/Q^3, Q^4/Q^2, and Q^3/Q^1 elements. Again, the MOC1 scheme was used for $Re = 5000$. Results essentially coincided with those in Figs. 17 and 18. To exaggerated discontinuities in the pressure, a coarser contour gridding was used and the resulting pressure contours are shown in Fig. 20.

7.5 Three–Dimensional Driven Cavity

A three-dimensional version of the driven cavity problem was also tested and results are shown in Fig. 21.

A relatively crude mesh of 22 cubic elements (Q^3/Q^1) with one level of h-refinement is shown in Fig. 21. This corresponded to a model with only 2244 degrees of freedom. A uniform velocity $u_2 = 1.0$ is presented over the lid face ($x = 0$) and the RK4 scheme is used to integrate the Navier-Stokes equations. In this case, $Re = 100$ and $\Delta t = 0.1$ sec. The

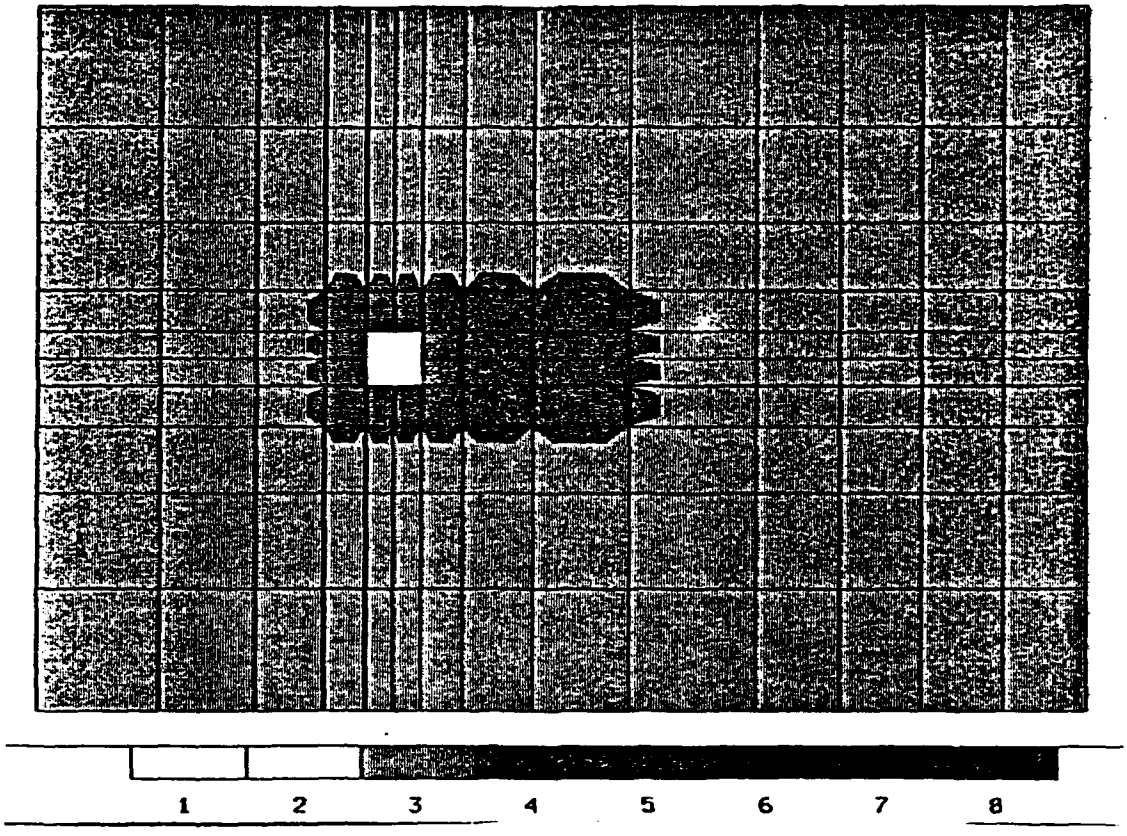
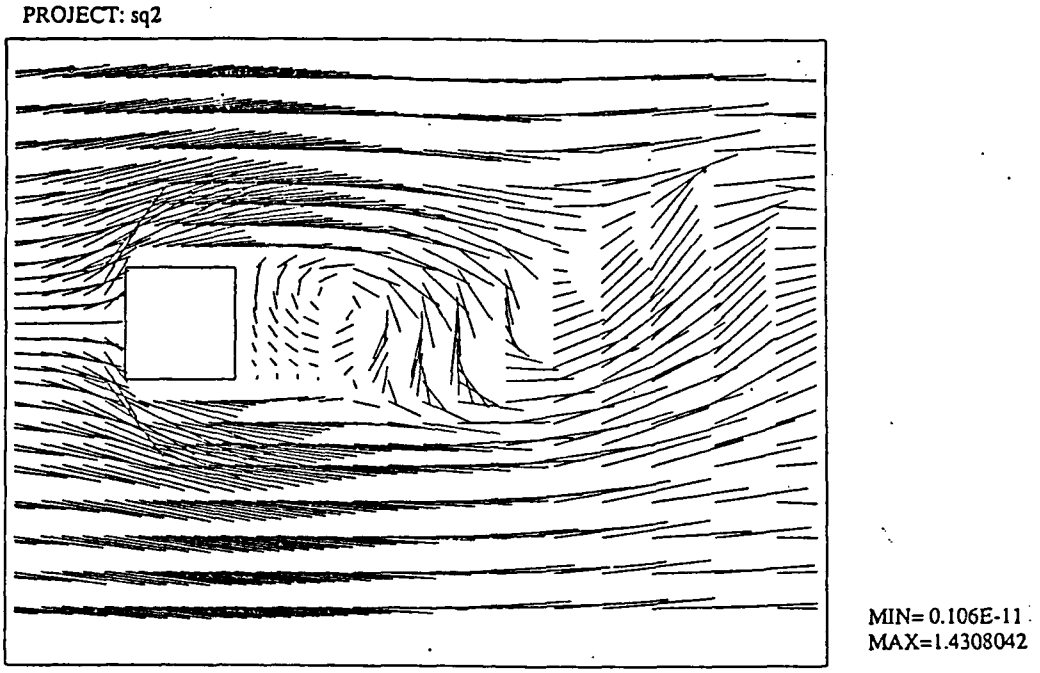
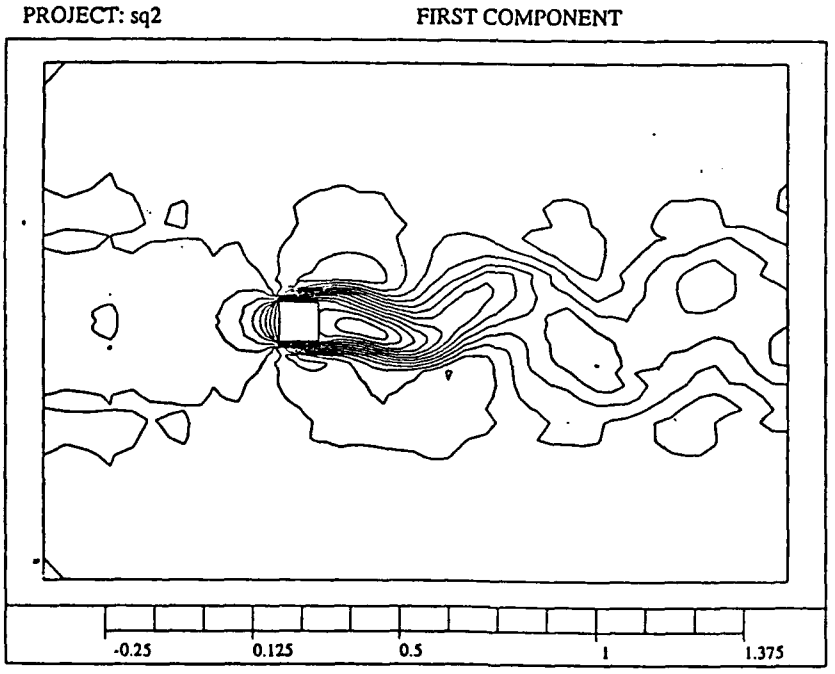


Figure 13: *hp* mesh for simulation of flow past a square obstacle; $Re = 250$.

Velocity vector magnitudes.





Contours of u component of the velocity.

Figure 14: Computed velocity field and contours of u_1-velocity at $t = 230$ sec., using Runge-Kutta method.

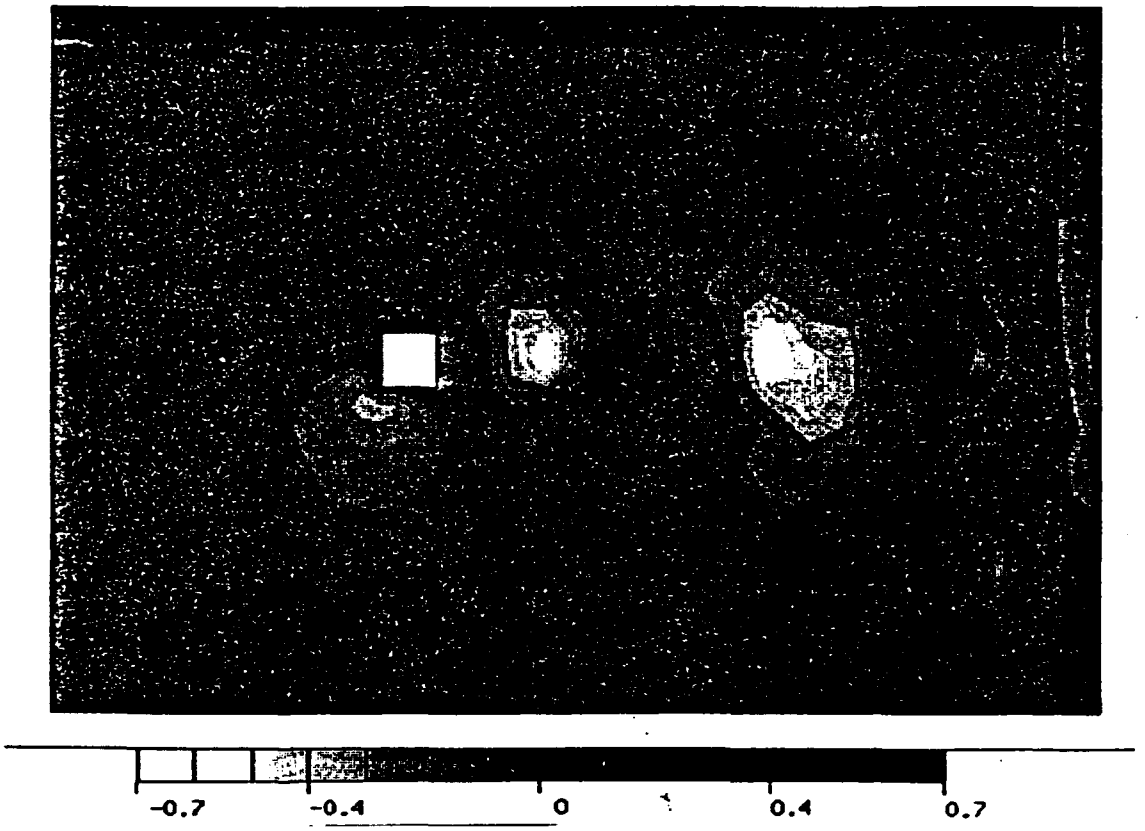


Figure 15: Computed u_2-velocity contours at $t = 230$ sec., $Re = 250$, Runge-Kutta method.

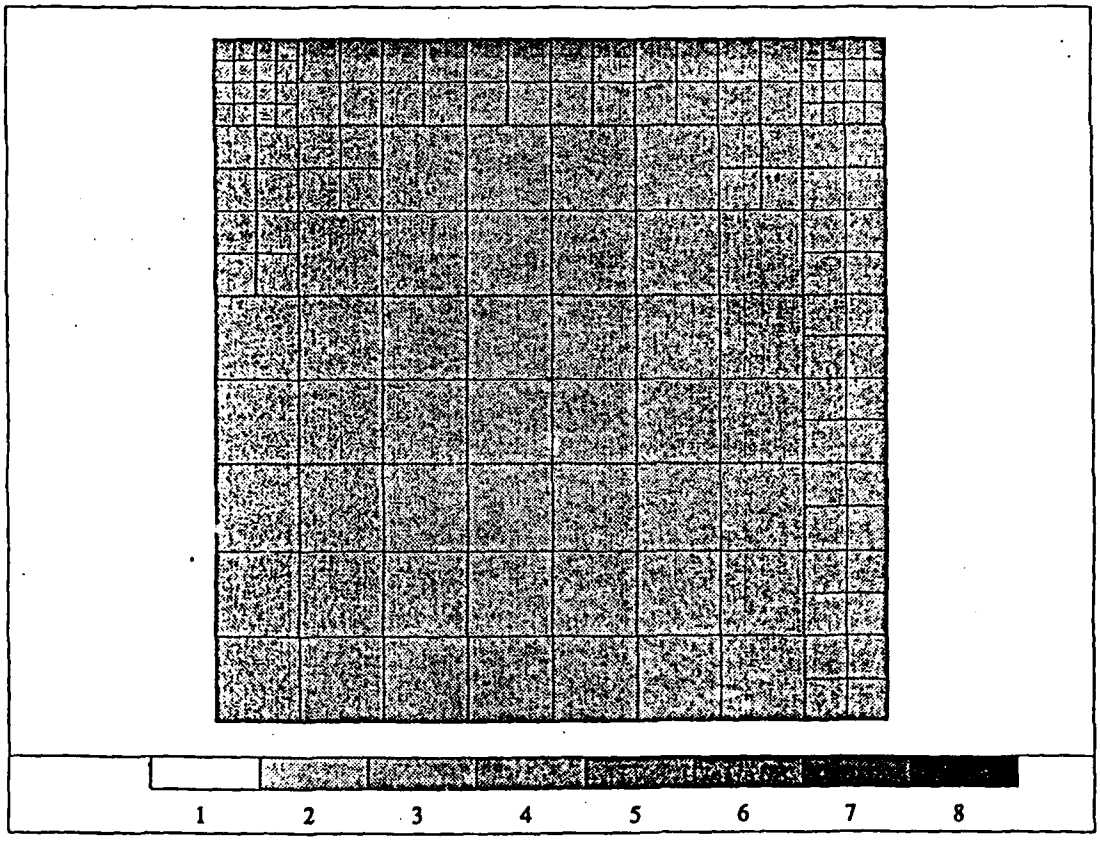
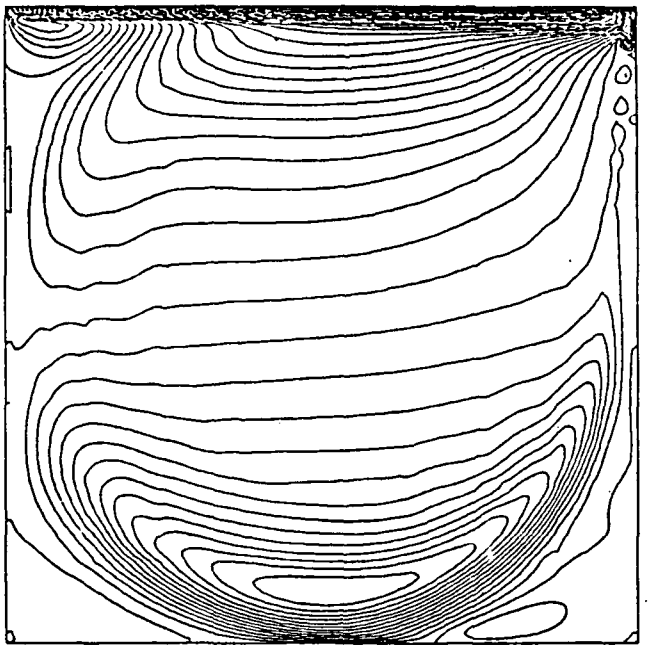
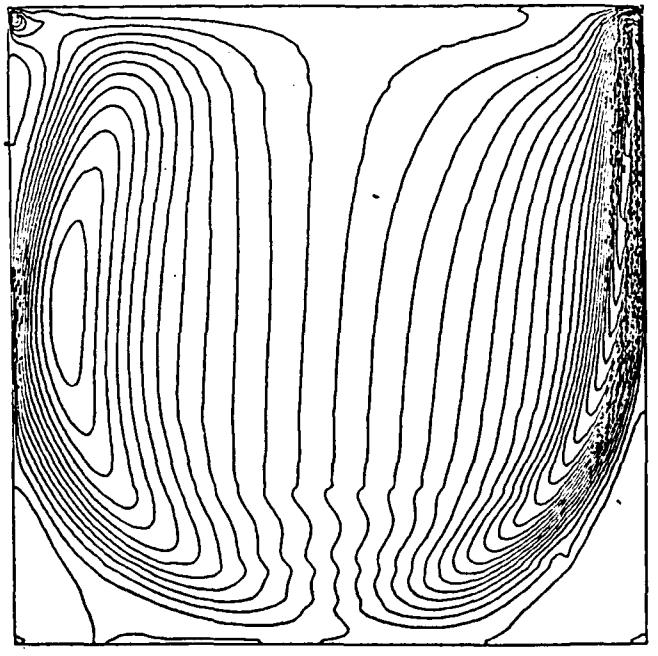


Figure 16: An *hp* mesh for the classical driven cavity problem, with adapted h-refinements and $p = 4$.

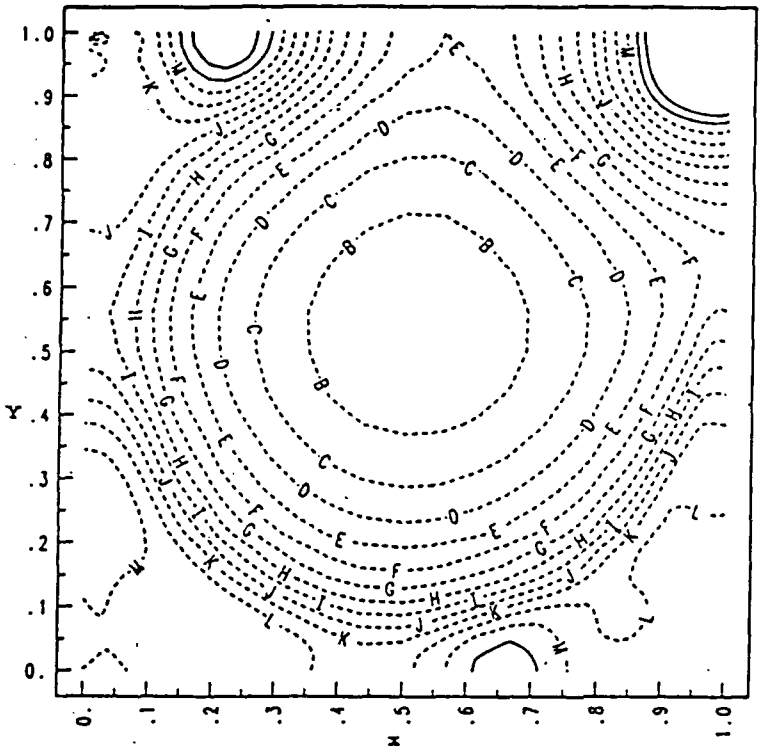


u velocity contours



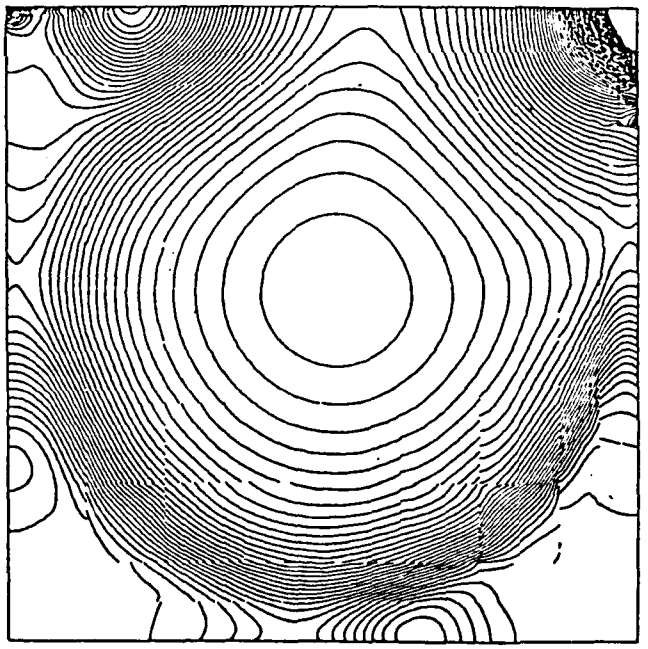
v velocity contours

Figure 17: Computed $u_1 = u$ and $v_2 = v$ velocity contours for $Re = 5000$, method of characteristics.



Pressure

(Gresh & Chan, 1987)



Pressure

hp (penalty method + m.o.c.)

Figure 18: Computed pressure contours compared with those of Gresho and Chan [58].

computed profiles at $t = 10.0\ \Delta t$ are shown superimposed on the mesh. A detailed discussion of results obtained on this and other three–dimensional benchmark problems is forthcoming.

8 References

1. DEMKOWICZ, L., ODEN, J. T., and STROUBOULIS, T., "An Adaptive p-Version Finite Element Methods for Transient Flow Problems with Moving Boundaries," **Proceedings, Finite Elements in Flow Problems**, Austin, 1983 and in **Finite Elements in Fluids**—Volume 6, Edited by R. H. Gallagher, G. F. Carey, J. T. Oden, and O. C. Zienkiewicz, John Wiley and Sons, Ltd., Chichester, pp. 291–305, 1984.

2. DEMKOWICZ, L., ODEN, J. T., and STROUBOULIS, T., "Adaptive Finite Element Methods for Flow Problems with Moving Boundaries, 1. Variational Principles and *A Posteriori* Estimates," *Computer Methods in Applied Mechanics and Engineering*, Vol. 46, No. 2, pp. 217–251, 1984.

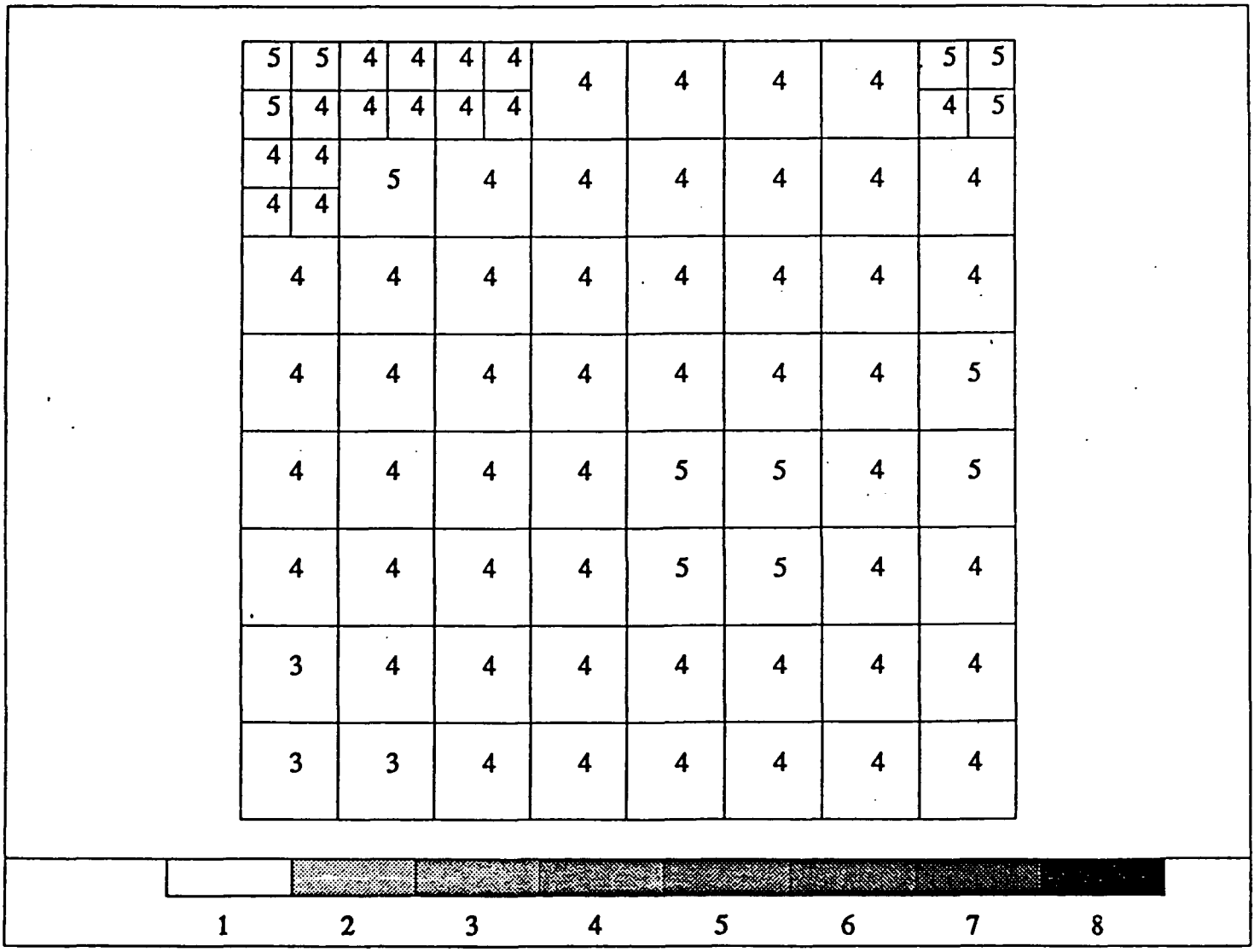
3. ODEN, J. T., DEMKOWICZ, L., STROUBOULIS, T., and DEVLOO, P., "Adaptive Methods for Problems in Solid and Fluid Mechanics," **Accuracy Estimates and Adaptive Refinements in Finite Element Computations**, Edited by I. Babuška,, O. C. Zienkiewicz, J. Gago, and E. R. A. de Oliveira, John Wiley and Sons, Ltd., London, 1986.

4. ODEN, J. T., STROUBOULIS, T., and DEVLOO, P., "Adaptive Finite Element Methods for the Analysis of Inviscid Compressible Flow: I. Fast Refinement/Unrefinement and Moving Mesh Methods for Unstructured Meshes," *Computer Methods in Appl. Mech. and Engrg.*, Vol. 59, No. 3, 1986.

5. ODEN, J. T., STROUBOULIS, T., and DEVLOO, P., "Adaptive Finite Element Methods for Compressible Flow Problems," **Numerical Methods for Compressible Flows—Finite Difference, Element and Volume Techniques**, Edited by T. E. Tezduyar and T. J. R. Hughes, AMD–Vol. 78, ASME, New York, pp. 115–126, 1987.

6. ODEN, J. T., STROUBOULIS, T., and DEVLOO, P., "Adaptive Finite Element Methods for High–Speed Compressible Flows," *International Journal for Numerical Methods in Fluids*, Vol. 7, pp. 1211–1228, 1987.

7. DEVLOO, P., ODEN, J. T., and PATTANI, P., "An Adaptive h-p Finite Element Method for Complex Compressible Viscous Flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 70, pp. 203–235, 1988.



D.O.F= 1442

Figure 19: Nonuniform hp mesh: $Re = 5000$.

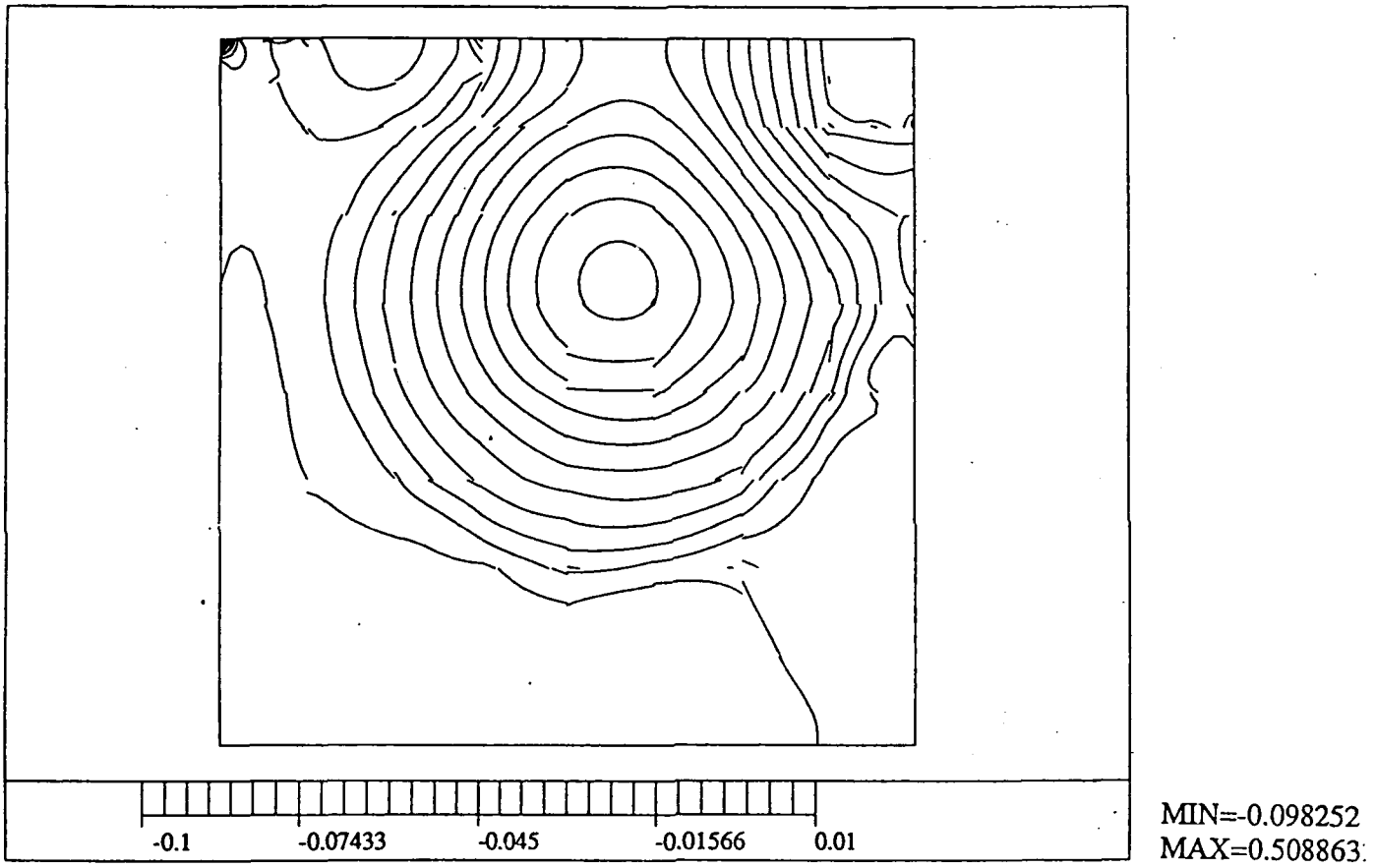


Figure 20: Coarse contours of pressure showing discontinuous but stable pressures on a nonuniform *hp* mesh.

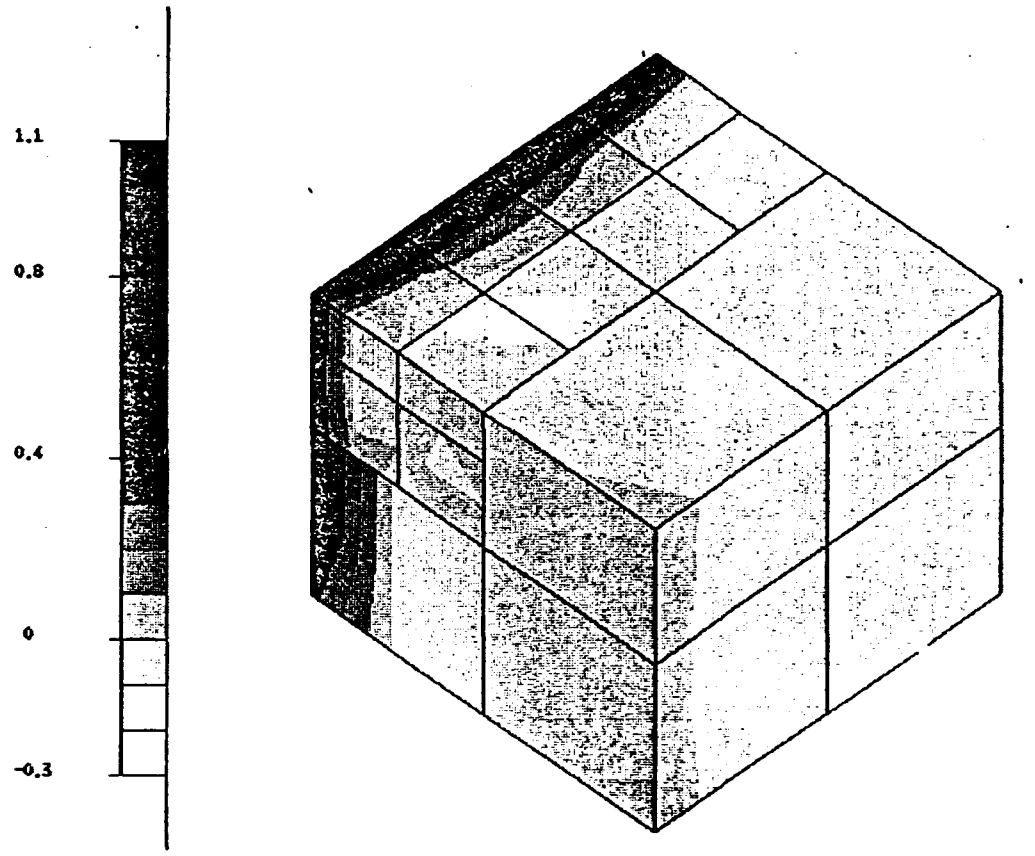


Figure 21: u-component velocity contours for an h-adapted mesh of cubic elements for the three–dimensional driven cavity problem: $Re = 100$, 2244 degrees of freedom.

8. DEMKOWICZ, L., ODEN, J. T., and RACHOWICZ, W., "A New Finite Element Method for Solving Compressible Navier-Stokes Equations Based on an Operator Splitting Method and *h-p* Adaptivity," *Computer Methods in Applied Mechanics and Engineering* (to appear).

9. ODEN, J. T., BASS, J. M., HUANG, C. Y., and BERRY, C. W., "Recent Results on Smart Algorithms and Adaptive Methods for Two- and Three–Dimensional Problems in Computational Fluid Mechanics," *Computers and Structures*, Vol. 35, No. 4, pp. 381–396, 1990.

10. BABUŠKA, I., ZIENKIEWICZ, O. C., GAGO, J., and OLIVEIRA, E. R. A. (Eds.), **Accuracy Estimates and Adaptive Refinements in Finite Element Computations**, John Wiley and Sons, Ltd., Chichester, 1986.

11. ODEN, J. T., and DEMKOWICZ, L., "Advances in Adaptive Improvements: A Survey of Adaptive Methods in Computational Fluid Mechanics," **State of the Art Surveys in Computational Mechanics**, Edited by A. K. Noor and J. T. Oden, American Society of Mechanical Engineers, N.Y., 1988.

12. ODEN, J. T., "Adaptive Finite Element Methods for Problems in Solid and Fluid Mechanics," **Finite Element Theory and Application Overview**, Edited by R. Voight, Springer-Verlag, N.Y., 1988.

13. ODEN, J. T., "Adaptive FEM in Complex Flow Problems," **The Mathematics of Finite Elements with Applications**, Edited by J. R. Whiteman, London Academic Press, Lt., Vol. 6, pp. 1–29, 1988.

14. ODEN, J. T., "Smart Algorithms and Adaptive Methods in Computational Fluid Dynamics," **Proceedings CANCAM (Canadian Congress on Applied Mechanics)**, Ottowa, Canada, 1989.

15. ODEN, J. T., "Progress in Adaptive Methods in Computational Fluid Dynamics," **Adaptive Methods for Partial Differential Equations**, Ed. by J. Flaherty, et al., SIAM Publications, Philadelphia, 1989.

16. ODEN, J. T., and BASS, J. M., "New Developments in Adaptive Methods for Computational Fluid Dynamics," **Computing Methods in Applied Sciences and Engineering**, Edited by R. Glowinski and A. Lichnewsky, SIAM Publications, Philadelphia, pp. 180–210, 1990.

17. BABUŠKA, I., and SURI, M., "The *p*- and *h-p* Versions of the Finite Element Method, An Overview," *Computer Methods in Applied Mechanics and Engincering*, Vol. 80, pp. 5–26, 1990.

18. ODEN, J. T., "Finite Elements: An Introduction," **Handbook of Numerical Analysis**, Edited by J. L. Lions and P. G. Ciarlet, Volume II, Part 1, Finite Elements, North-Holland, Amsterdam, pp. 3–15, 1991.

19. RITZ, W., *Journal für der Reine und Angewandte Mathematik*, Vol. 135, 1908.

20. GALERKIN, B. G., "Rods and Plates," vestnik inženerov, **19** (in Russian), 1915.

21. KANTOROVICH, L. V., and KRYLOV, V. I., **Approximate Solutions of Partial Differential Equations**, Moscow, Leningrad (in Russian), 1936.

22. ODEN, J. T., "Analysis of Plate–Beam Structures," *Ph.D. Dissertation*, Okla. St. Univ., Stillwater, 1962.

23. SZABÓ, B., and MEHTA, A. K., "*P*-Convergent Finite Element Approximations in Fracture Mechanics," *International Journal for Numerical Methods in Engineering*, Vol. 12, pp. 551–560, 1978.

24. BABAŠKA, I., SZABÓ, B. A., and KATZ, I. N., "The *p*-Version of the Finite Element Method," *SIAM J. Numerical Analysis*, **18**, pp. 512–545, 1981.

25. SZABÓ, B. A., "Mesh Design for the *p*-Version of the Finite Element Method," *Computer Methods in Applied Mechanics and Engineering*, **55**, pp. 181–197, 1986.

26. SZABÓ, B. A., "Estimation and Control of Error Based on *P*-Convergence," **Accuracy Estimates and Adaptive Refinements in Finite Element Computations**, Edited by I. Babuška, J. Gago, E. R. de A. Oliveira, and O. C. Zienkiewicz, John Wiley and Sons, Ltd., Chichester, pp. 61–78, 1986.

27. SZABÓ, B., "The *p*- and the *h*-*p* Versions of the Finite Element Method in Solid Mechanics," *Computer Methods in Applied Mechanics and Engineering*, **80**, pp. 185–195, 1990.

28. SZABÓ, B. A., "The Use of *A Priori* Estimates in Engineering Computations," **Reliability in Computational Mechanics**, Edited by J. T. Oden, North-Holland, Amsterdam, pp. 139–154, 1990.

29. BABUŠKA, I, and DORR, M. R., "Error Estimates for the Combined *h*- and *p*-Version of the Finite Element Method," *Numer. Math.*, **37**, pp. 257–277, 1981.

30. BABUŠKA, I, "The *p* and *h*-*p* Versions of the Finite Element Method. The State of the Art," in D. L. Dwoyer, M. Y. Hussaini, and R. G. Voigt (Eds.), *Finite Elements Theory and Application*, pp. 199–239, Springer-Verlag, New York, 1988.

31. BABUŠKA, I, and SURI, M., "The Optimal Convergence Rate of the *p*-Version of the Finite Element Method," *SIAM J. Numer. Anal.*, **24**, pp. 750–776, 1987.

32. BABUŠKA, I, and SURI, M., "The *h-p* Version of the Finite Element Method with Quasi–Uniform Meshes," *Math. Model. Numer. Anal. (RAIRO)*, **21**, pp. 199–238, 1987.

33. GUO, B., and BABUŠKA, I, "The *h-p* Version of the Finite Element Method, Part I, The Basic Approximation Results; Part II, General Results and Applications," *Comput. Mech.*, **1**, pp. 21–41, pp. 203–226, 1986.

34. BABUŠKA, I,and SURI, M., "The *p*-Version of the Finite Method for Constraint Boundary Conditions," *Math. Comp.*, **51**(1873), pp. 1–13, 1988.

35. BABUŠKA, I, and GUO, B. Q., "The *h-p* Version of the Finite Element Method for Problems with Nonhomogeneous Essential Boundary Condition," *Comp. Methods Appl. Mech. Engrg.*, **74**, pp. 1–28, 1989.

36. PATERA, A. T., "A Spectral Method for Fluid Dynamics, Laminar Flow in a Channel Expansion," *J. Computational Physics*, **54**, pp. 468–488, 1984.

37. RONQUIST, E. M., "Optimal Spectral Element Methods for the Unsteady Three–Dimensional Incompressible Navier-Stokes Equations," *Ph.d. Dissertation*, M.I.T., Cambridge, 1988.

38. MADAY, Y., and PATERA, A. T., "Spectral Element Methods for the Navier-Stokes Equations," **State-of-the-Art Surveys in Computational Mechanics**, Edited by A. K. Noor, and J. T. Oden, ASME, NY, 1989.

39. HO, L-W, and PATERA, A. T., "A Legendre Spectral Element Method for Simulation of Unsteady Incompressible Viscous Free–Surface Flows," *Computer Methods in Applied Mechanics and Engineering*, **80**, pp. 355–366, 1990.

40. MAVRIPILIS, K., "Nonconforming Discretizations and *A Posteriori* Error Estimates for Adaptive Spectral Element Techniques," *Ph.D. Dissertation*, M.I.T., Cambridge, 1989.

41. DEMKOWICZ, L., ODEN, J. T., RACHOWICZ, W., and HARDY, O., "Toward a Universal *h-p* Adaptive Finite Element, Part 1. Constrained Approximation and Data Structures," *Computer Methods in Applied Mechanics and Engineering*, Vol. 77, pp. 79–112, 1989.

42. ODEN, J. T., DEMKOWICZ, L., WESTERMANN, T., and RACHOWICZ, W., "Toward a Universal *h-p* Adaptive Finite Element Strategy, Part 2. *A Posteriori* Error Estimates," *Computer Methods in Applied Mechanics and Engineering*, Vol. 77, pp. 113–180, 1989.

43. RACHOWICZ, W., ODEN, J. T., and DEMKOWICZ, L., "Toward a Universal *h-p* Adaptive Finite Element Strategy, Part 3. A study of the Design of *h-p* Meshes," *Computer Methods in Applied Mechanics and Engineering*, Vol. 77, pp. 181–212, 1989.

44. GUNZBURGER, M. D., **Finite Element Methods for Viscous Incompressible Flows, A Guide to Theory, Practice, and Algorithms**, Academic Press, San Diego, 1989.

45. TEMAM, R., **Navier-Stokes Equations and Nonlinear Functional Analysis**, SIAM, Philadelphia, 1983.

46. SZEGO, G., *Orthogonal Polynomials*, American Math. Soc. Colloq. Publication, Vol. 23, Am. Math. Soc., Providence, R.I., 1939.

47. BABUŠKA, I., ELMAN, H. C., and MARKLEY, K., "Parallel Implementation of the *hp*-Version of the Finite Element Method on a Shared–Memory Architecture," *Institute for Advanced Computer Studies Computer Science Technical Report Series*, TR-90-139, Univ. of Maryland, 1990.

48. LADYSZHENSKAYA, O. A., **The Mathematical Theory of Viscous Incompressible Flow**, Gordon and Breach, NY, 1969.

49. CAREY, G. F., and ODEN, J. T., **Finite Element: A Second Course**, Vol. II, Prentice-Hall, Englewood Cliffs, 1983.

50. KIKUCHI, N., and ODEN, J. T., **Contact Problems in Elasticity**, SIAM Publications, Philadelphia, 1988.

51. BREZZI, F., and FORTIN, M., **Mixed Finite Element Methods** (manuscript, to appear).

52. ODEN, J. T., **Qualitative Methods in Nonlinear Mechanics**, Prentice-Hall, Englewood Cliffs, 1986.

53. WU, W., and ODEN, J. T., "Stability of Mixed and Penalty Approximations for High–Order Finite Element Approximation of the Stokes Problem in Two Dimensions," **TICOM Report**, TR-3-91, Austin, 1991.

54. CHORIN, A., "The Numerical Solution of the Navier-Stokes Equations for an Incompressible Fluid," *Bulletin of the American Math. Soc.*, Vol. 73, No. 6, p. 928, 1967.

55. PATANKAR, W. V., and SPALDING, D. B., "A Calculation Procedure for Heat, Mass, and Momentum Transfer in Three–Dimensional Parabolic Flows," *International J. of Heat and Mass Transfer*, Vol. 15, pp. 1787–1806, 1972.

56. CARETTO, L. S., GOSMAN, A. D., PATANKAR, S. V., and SPALDING, D. B., "Two Calculation Procedures for Steady, Three–Dimensional Flows with Recirculation," *Lecture Notes in Physics*, Springer-Verlag, NY, pp. 60–68, 1972.

57. GRESHO, P. M., "Incompressible Fluid Dynamics: Some Fundamental Issues," Lawrence Livermore National Laboratory Report UCRL-JC-104068, June 1990; also **Annual Review of Fluid Mechanics**, Vol. 23, January, 1991.

58. GRESHO, P. M., and CHAN, S. T., "On the Theory of Semi–Implicit Projection Methods for Viscous Incomprssible Flow and its Implementation Via a Finite Element Method that Introduces a Nearly–Consistent Mass Matrix," *International Journal for Numerical Methods in Fluids*, **11**, pp. 587–660, 1990.

59. RAMASWAMY, B., JUE, T. C., and AKIN, J. E., "A Review of Some Finite Element Methods to Solve the Incompressible Navier-Stokes Equations," *Comuter Methods in Applied Mechanics and Engineering* (to appear).

60. MARCHUK, G. I., "Splitting and Alternating Direction Methods," **Handbook of Numerical Analysis**, Edited by P. G. Ciarlet and J. L. Lions, North-Holland, Amsterdam, Vol. 1, Part 1, pp. 197–462, 1990.

61. CANUTO, C., HUSSANI, M. Y., QUARTERONI, A., and ZANG, T. A., **Spectral Methods in Fluid Dynamics**, Springer-Verlag, NY, 1987.

62. KOROCZAK, K. Z., and PATERA, A. T., "An Isoparametric Spectral Element Method for Solution of the Navier-Stokes Equations in Complex Geometry," *Journal of Computational Physics*, **62**, pp. 361–382, 1986.

63. BUTCHER, J. C., **The Numerical Analysis of Ordinary Differential Equations**, John Wiley and Sons, NY, 1987.

64. GHIA, K. N., OSWALD, G. A., and GHIA, U., "Analysis of Incompressible Massively Separated Viscous Flows Using Unsteady Navier-Stokes Equations," *Internationa Journal of Numerical Methods in Fluids*, Vol. 9, pp. 1025–1050, 1989.

65. STROUBOULIS, T., and ODEN, J. T., "*A Posteriori* Error Estimates for Problems in Computational Fluid Dynamics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 78, pp. 201–242, 1990.

66. ODEN, J. T., DEMKOWICZ, L., RACHOWICZ, W., and WESTERMANN, T. A., "*A Posteriori* Error Analysis in Finite Elements: The Element Residual Method for Symmetrizable Problems with Applications to Compressible Euler and Navier-Stokes Equations," **Reliability in Computational Mechanics**, Edited by J. T. Oden, and *Computer Methods in Applied Mechanics and Engineering*, Vol. 82, Nos. 1–3, pp. 183–203, 1990.

67. AINSWORTH, M., and ODEN, J. T., "A Unified Approach to *A Posteriori* Error Estimation Using Element Residual Methods," **TICOM Report**, TR-91-4, Austin, 1991.